**VŠB - Technical University of Ostrava**

**Faculty of Electrical Engineering and Computer Science**

**Department of Computer Science**

# Využití technologie SharePoint pro školní projekty

# Using the Technology SharePoint for School projects

**Martin Bartek** **2011**

Hereby I declare that this diploma thesis was written on my own. I have quoted all references I have drawn upon.

Ostrava 22th July 2011

I would like to thank ing. Jan Martinovič Ph.D. for the help and huge portion of patience.

# Abstrakt

Práce je zaměřená na představení technologie SharePoint a jejího nasazení. Představení základních znalostí o této technologii, určíme hardwarové a licenční požadavky systému a způsob nasazení. Prozkoumáme uživatelské rozhraní a základní architektonické prvky systému. Součástí práce je vypracování několika ukázkových aplikací využívajících SharePoint API a Web Part která zapadá do logiky projektu.

Projekty jsou kombinace desktopových aplikací a Windows služeb, komunikujících přes Windows Communication Foundation s Windowsovskou službou na serverové straně. Tato služba provádí veškeré operace s SharePointem. První projekt je určený pro vytváření nových kolekcí webů a druhá pro synchronizaci systémových složek s knihovnami na SharePointu a vytváří zálohy změněných souborů. Moje webová část (web part) spravuje tyto zálohy a dovoluje uživateli s nimi nakládat.

# Klíčová slova

SharePoint, Web Part, Windows service, WCF

# Abstract

The thesis is aimed at studying SharePoint technology and its deployment. I am going to introduce basic knowledge about this technology, define hardware and license requirements of the system and a way of deployment, examine user interface and base architectural elements of the system. This thesis includes implementing of several sample projects using SharePoint API and web part setting the overall project logic.

Projects are combination of desktop applications and Windows services communicating over a Windows Communication Foundation with Windows service on the server side. This service carries out all operations with SharePoint. The first project is designed to create new site collections and the second one to synchronize the system folders with libraries in SharePoint and creates backups of changed files. My Web Part manages these backups and allows user to manage them.

# Keywords

SharePoint, Web Part, Windows service, WCF

# Content

1. Thesis structure ..... 8
2. Motivation ..... 9
3. SharePoint ..... 10
3.1. Which features we can find in basic SharePoint Foundation 2010 ..... 11
3.2. What features are in addition in full value SharePoint Server 2010? ..... 12
3.3. What is new in SharePoint Foundation 2010 (SPF 2010) in comparison with WSS? ..... 13
3.4. SharePoint Basics ..... 16
3.4.1. Sites ..... 16
3.4.2. SharePoint farm ..... 16
3.4.3. SharePoint site structure ..... 17
3.4.4. Permissions ..... 17
3.4.5. Web Parts ..... 18
3.4.6. Web Page ..... 19
3.4.7. Lists ..... 20
3.4.8. Document libraries ..... 20
3.4.9. Interface ..... 20
3.4.10. SharePoint Central Administration ..... 23
3.4.11. Site Settings ..... 24
4. How to create a new site in SharePoint environment ..... 26
5. Implementation ..... 28
5.1. Before you start ..... 28
5.1.1. SharePoint 2010 64 bit ..... 28
5.1.2. SharePoint 2010 .Net Framework ..... 28
5.1.3. SharePoint and remote server ..... 28
5.2. Implementation technologies and approaches ..... 28
5.2.1. Projects' general communication architecture ..... 29
5.2.2. Windows Communication Foundation ..... 29

5.2.3. Web Parts developing ..... 30
5.2.4. Windows Service Programming ..... 31
5.3. Implementation issues ..... 34
5.3.1. SharePoint installation on Windows 7 client OS ..... 34
5.3.2. Security problem – cannot modify/create Web applications on client OS ..... 34
5.3.3. SharePoint is accessible, but your credentials are not accepted ..... 35
5.3.4. Access to SharePoint from Internet when on virtual machine ..... 35
5.3.5. Access to SharePoint when you are behind a home router ..... 35
5.3.6. Testing problem ..... 36
5.3.7. Windows Installer setup ..... 37
6. Projects ..... 38
6.1. SharePoint Site Creator ..... 38
6.1.1. User guide ..... 38
6.1.2. Implementation ..... 41
6.1.3. Project conclusion ..... 43
6.2. SharePoint File Synchronizer ..... 43
6.2.1. User guide ..... 44
6.2.2. Implementation ..... 47
6.2.3. Project conclusion ..... 50
6.3. BackupViewer Web Part ..... 51
6.3.1. User guide ..... 51
6.3.2. Implementation ..... 52
6.3.3. Project conclusion ..... 52
7. Conclusion ..... 53
8. Bibliography ..... 54
9. Appendix ..... 56
9.1. Licenses and proper version choice ..... 56
9.2. Requirements ..... 57
9.3. Server pre-install preparations ..... 58

9.4. SharePoint installation..... 58

# 1. Thesis structure

In this thesis I am going to explain why I have chosen SharePoint as a topic (Chapter 2). Then the real work begins and I am going to introduce SharePoint (Chapter 3), tell what versions we can choose and compare them and their differences (Chapter 3.1, Chapter 3.2) and what features a new SharePoint 2010 version brought to users (Chapter 3.3). Then I plan to introduce SharePoint site structure and basics (Chapter 3.4) important to its use like basic objects, interface and settings and to describe how to create new site collections for our projects (Chapter 4).

Then the practical part comes up and I will tell you what you should know before you start to implement any project (Chapter 5.1). When I designed and developed my project, I had to use range of techniques and technologies and I would like to describe some of the most important ones (Chapter 5.2).

When I was preparing my test servers and during implementing I had to face many problems and issues. Solving some of them took few minutes, but other ones were quite tough and it took even days to find a way how to solve them and I could go on. Sometimes I had a feeling that I gathered all possible problems and when I solved one, other 2 appeared and some of them were rare. I chose to describe the ones that could be useful even for other people (Chapter 5.3). And now the project part comes, describing project goals and contains project introduction, user guides, and implementation and concludes each of these projects (Chapter 6).

That could be the end of my thesis. It is finished by conclusion of whole diploma paper (Chapter 7) followed by a list of sources (Chapter 8) and supplement (Chapter 9). The appendix contains some interesting information such as licenses, requirements, server preinstall preparations and SharePoint installation that could be useful to everybody who wants to deploy his own SharePoint server.

# 2. Motivation

When I was choosing my diploma thesis, I wanted something new - some new technology, which could be useful in my future professional life and SharePoint and Customer Care Framework came on my mind. Both were quite widespread in software development branch and a knowledge that I could learn would have a price of gold in a future.

So I chose Customer Care Framework – a framework made mainly for Call Center development. I had some connections with people from the branch and I could test my software on actual Call Center solutions.

But the crisis came and I have lost those connections and deployment testing would be almost impossible or very difficultly practicable. Another misfortune came when Microsoft has decided to cancel CCF as a standalone project and has made a plugin of it for Customer Relationship Management 4.0.

At that moment I have decided to abandon my original thesis topic and have come with a new idea – to study SharePoint, which was second on my mind in a time of decision. And I think that was a good idea. SharePoint is an extremely useful technology these days and I am glad I have chosen it instead of plugin for CRM.

But if you are interested, you can find the original unfinished project on a disk. It is about 25 pages of theory about Call Centers in English.

# 3. SharePoint

When you create some project and you are not completely alone who creates it, it is necessary to cooperate with another people – your co-workers – to complete the task. You have to share information, send documents and manage versions of all these documents and source codes to prevent chaos in your effort. And more people usually mean more problems and your collaboration becomes more and more complex and frustrating.

That is generally why the SharePoint was made – to ensure effective cooperation among all employees and thus ease project development and save company resources (both human and financial). Nowadays the SharePoint or any other competitive technology is a standard component of company's software development tools.

Because of these tools, it is easily possible to assign tasks within a team and watch its advance and share data and documents within the whole project, so every employee knows what a team goal is and what his position is in the team. And all of that happens within a corporate intranet. Workers can easily communicate and discuss problems and their possible solutions.

And that is SharePoint. A collaboration platform within company or group that allows creating, managing, searching and accessing shared information and files, check workflows, manage security and storage for many people (see Figure 1). To do so SharePoint provides many predefined classes and structures. SharePoint developers can use extensive API for their applications and SharePoint was designed to cooperate with Microsoft's products.


**Figure 1 - SharePoint features [1]**

Original SharePoint was developed already in 2001 under the name SharePoint Team Services for easy team webs creating, which primarily included project documents, task lists and calendars. The name was later changed to Windows SharePoint Services (WSS) and even usage began to expand further from software development. Users started to use it for resource management and customer service assistance.

Basic edition is called SharePoint Foundation 2010, which is possible to download completely for free from Microsoft sites and it is not necessary to pay for any other licenses. This free version was originally called SharePoint Services (WSS) and it was a component of Windows Servers 2008, 2008 R2 and 2003. SharePoint Foundation 2010, the newest version is not yet a component of any Windows Server version, but we can expect it in new OSs when released.

SharePoint Foundation 2010 is mainly destined for smaller companies and individuals that really appreciate cost reduction and make do with basic tool set and features like time schedule coordination, document sharing and team communication through workspaces, blogs and document libraries [1]. It is good choice for people who start with SharePoint and want to learn it and test whether they really need it.

Another option is SharePoint Sever 2010, which is destined mostly for larger companies which can afford commercial licensees and enjoy an advantage of better functionality compared to SharePoint Foundation.

SharePoint Server 2010 allows team leaders to connect people by creating interest communities, to react easier for change of company requests based on data-drive decisions and to help users to find needed data, information and people for their projects.

You can download SharePoint Foundation 2010 on Microsoft sites.[1]

## 3.1. Which features we can find in basic SharePoint Foundation 2010[2]

If you think that you don't need commercial version and a free version can completely satisfy your needs, because it has everything you require, you can save some money. Let's check what the basic tools are in SharePoint Foundation:

[1] Download your SharePoint here: http://www.microsoft.com/downloads/en/details.aspx?FamilyID=49c79a8a-4612-4e7d-a0b4-3bb429b46595&displaylang=en

[2] You can find another basic features at http://technet.microsoft.com/cs-cz/sharepoint/ee518670.aspx and http://www.networkworld.com/community/node/56630

| | |
|---|---|
| **Basic structures** | Structures like Web applications, Site Collection, Sites, Lists, Document libraries, web parts and web pages and many more.[3] |
| **Site management tool** | It is a collection of tools for editing and administering browser-based pages and collections of variety forms. |
| **Central Administration Console Tools** | Administrator tools for managing SharePoint server, backing up, data restoring etc. |
| **Business Connectivity Services** | It is a set of services and features that allows connecting SharePoint-based solutions to other systems and allows defining content-types on these external data basis. |
| **Health Monitoring** | Includes tools for health analysis, which checks possible issues in configuration, performance and usage. |

## 3.2. What features are in addition in full value SharePoint Server 2010?

SharePoint Server 2010 includes all things like Windows SharePoint Foundation 2010 and adds more features:

- Afford opportunity to users creating own Sites and modify personal information, which then can be used by whole organization within SharePoint system. That is important for project leaders during looking for the right people for their projects. These personal information are possible to connect to Active Directory and to create user databases includes metadata and then manage within SharePoint.
- Site Directory Feature is very useful component when you have very large amount of sites, which gathers on one place and allows adding them to categories and then sorting them by these categories.
- Allows search and index information and data even out of own databases – for example it allows to index web pages, Exchange public folders and shared files and other sources out of SharePoint.

[3] More detail description is in SharePoint Basics chapter

- Markedly extends workflow usage compared to SharePoint Foundation, where the usage is considerably shorted.
- Afford possibility integrating with Microsoft Information Right Management (IRM) – specify rights for document reading, marking what is actually in the document and what a user can do with this information [2].
- Allows restrict and analyze procedures and log all site, content and workflow actions.
- Allows to edit expiration times, watch and audit activities including creating security policies.
- Include integrated InfoPath for XML-based forms and allows conversion of these XML-based documents to HTML form.
- Allows create Excel services for work with tables.
- With using BDC (Business data catalog) allows to datamine external databases.
- Uses Single Sign-on technology.
- Includes considerable extended template options for Site creating.

## 3.3. What is new in SharePoint Foundation 2010 (SPF 2010) in comparison with WSS?[4]

When the newest SharePoint version was released in June 2010, it brought quite lot novelties (see Figure 2 and Figure 3). Some of them were only visual and some other ones were functional and had serious impact for a future development. It brought compatibility with many Microsoft products. You can see in the list or images, that number of changes is not small at all.

List of most important novelties:

- Alert function improvement. Now is possible to notice administrators with SMS.
- Business Connectivity Services provide read/write functionality to external sources such as databases, web services etc.
- Client Object Model: SPF2010 brings new APIs for interacting with SharePoint Sites - .NET Framework 3.5 or later, Microsoft Silverlight and JavaScript.
- Event improvements: There are several new events in SPF 2010 such as on-create and on-delete events for lists and Web Sites, site collection binding event, XML binding event and few more. Some of the events can be synchronous or asynchronous [3].

[4] You can find another new features at http://sharepoint.vanglabbeek.nl/index.php/2009/11/02/sharepoint-foundation-2010-whats-in-a-name/

- Microsoft Synch Framework: Synchronization with third-party application is easier. It is complex synchronization architecture providing data-agnostic and bidirectional capabilities to developers.
- Connectivity with mobile devices was greatly improved. SharePoint is compatible with Forefront Unified Access Gateway allowing simple access from anywhere by mobile devices such as smart phones with Windows Phone 7.
- New filtering and querying ways were added.
- Ribbon technology was implemented in SPF2010 GUI. If you like ribbons, you will like SPF2010 interface. Another interface design changes were made including revamped CSS and masterpage. Design of interface is lucid and makes it easier to work with objects.
- Sandbox solution availability is a new concept that allows site users to create their own solutions with their custom code and store them in a gallery. Each sandbox solution works in separated environment. Farm administrator can analyze them and decide about their usability, memory and CPU time consumption and check exceptions without endangering whole farm or sites.
- Service Application Framework is an API providing SharePoint Services functionality for new created applications to computers across the farm (or farms) and replacing Shared Service Provider from older versions. There are already more than 20 services implemented in SPF2010 (SharePoint Search for example).
- Even it was possible to host Silverlight applications in Web Parts, new version of SharePoint provides built-in, extensible Silverlight Web Parts made specifically for Silverlight applications. It could be hosted in Silverlight Web Part or Silverlight Cross Domain Data Access System, through which Silverlight is connected to SharePoint data.
- PowerShell command line tool was added to interface and new functions were added to PowerShell SharePoint library. Thanks to SharePoint it is easier to make scripts and automate often repeated procedures.
- In SPF2010 the Workflow was improved since SharePoint Services 3.0 allowing making richer and flexible workflows and business scenarios. Workflow activities (known as actions) build blocks. Number of events invoking by workflows was increased and now it is possible to make new custom events for them. New site workflows are independent on list items and can be associated with specified events and other workflow activities.
- There is a new remote BLOB storage device possibility for SQL server Express version enlargement.
- Editable shared documents – it is easily possible to work with one document with several people. Any changes are automatically replicated to all other open version of such document, so it is possible to watch which changes are made by your co-workers.

- Validation feature that validates inserting data and easily check formal and partially even logical correctness of recorded documents. That is because of evaluating, calculating and comparing validators.
- Now you can work offline with your documents. Any changes are synchronized when you are connected to the internet. So it is not necessarily to be online all the time. This feature is useful when traveling.
- SharePoint Foundation 2010 can be installed on client operation systems Windows Vista and Windows 7.
- Multilanguage support – the web interface supports multilingual mutations. So navigation, column names etc. are adapted to native language of developer's native language, if set up. That can be especially useful in multination corporations when project developers who work on one particular document can be from different countries.
- Pagination has an impact on database filtering. With previous versions if you read from database for example 100 items from a table, SharePoint read all data and then showed first 100 in the view on the first page. Now it reads only 100 items.
- SharePoint 2010 supports recycle bin. Items you remove are not permanently deleted, but moved to the bin (if enabled).

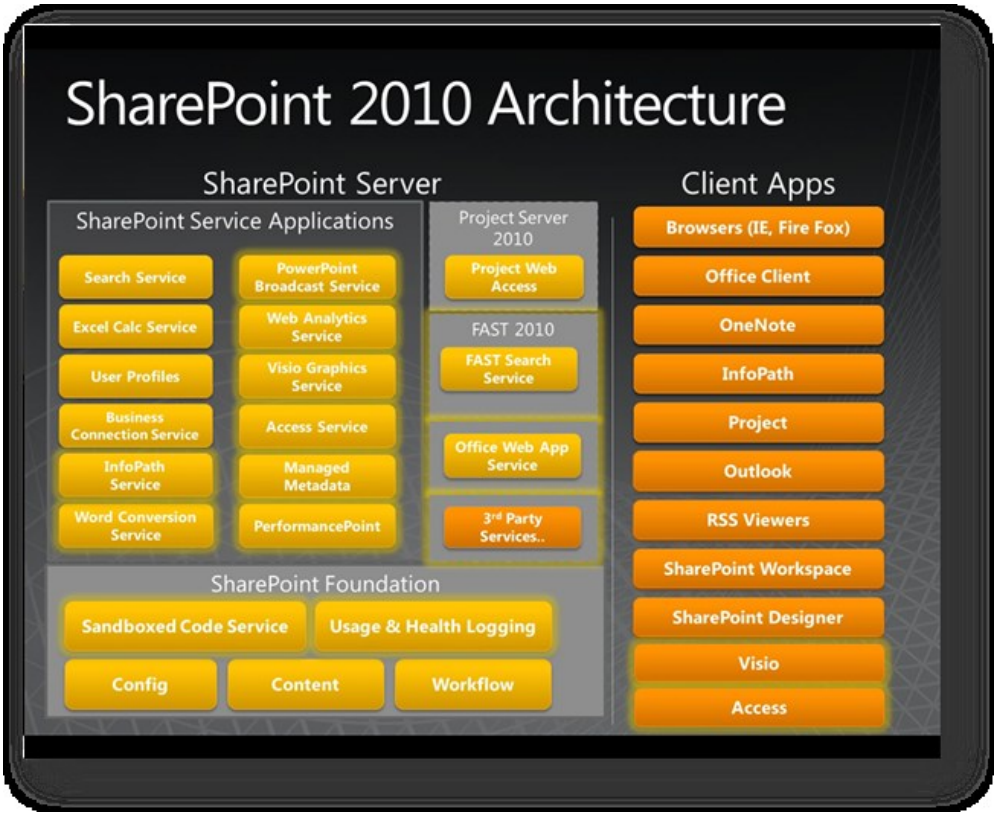


**Figure 2 - SharePoint 2010 Architecture [5]**

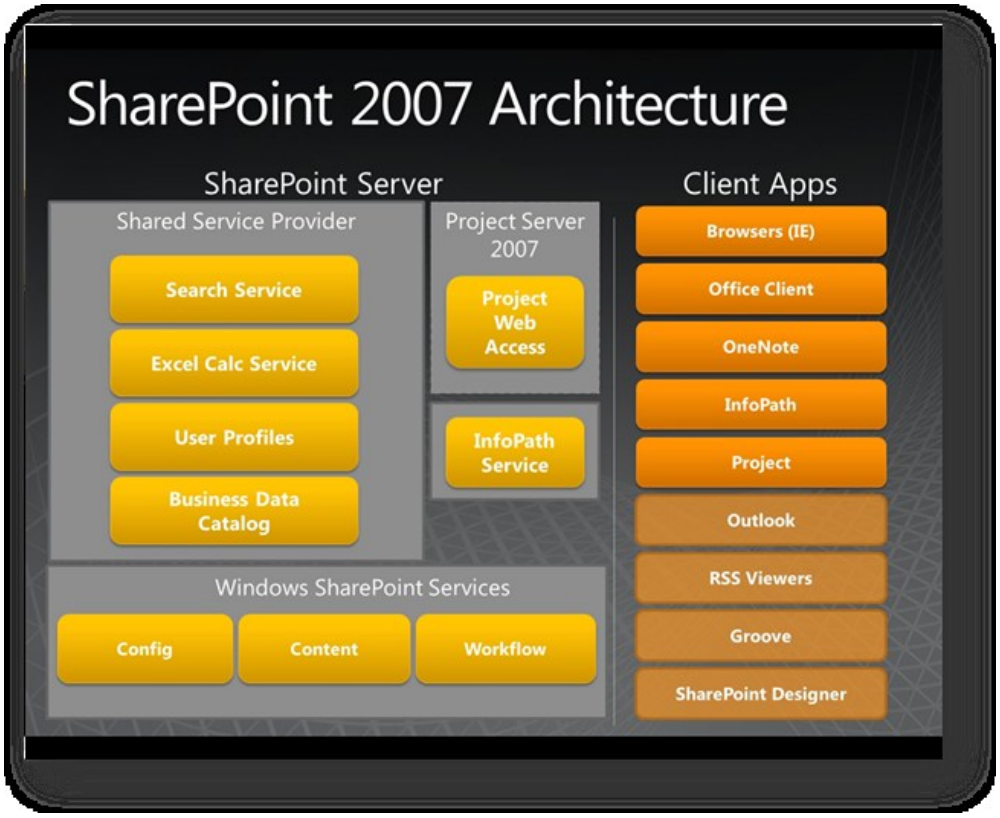


**Figure 3 - SharePoint 2007 Architecture [5]**

## 3.4. SharePoint Basics

There are a few things that we will need at the beginning for working with SharePoint. You can use most of today's browsers like Firefox 3.6 or later or Internet Explorer 7 or later.

The SharePoint product belongs to Office family product line (former name of SharePoint was Office SharePoint), so these Microsoft products are very close and support each other. Office software is an important ingredient for full SharePoint usage for information portal and is very common in business deployment. It is very appropriate to have Office 2007 version or later.

### 3.4.1. Sites

Site is a place for our projects. The place where we can share information, discuss problems and watch progress we made. We can create blogs and team discussion forums. It is the place for managing of all lists, document libraries and web pages.

### 3.4.2. SharePoint farm

A SharePoint farm is an environment that can include many types and numbers of servers providing SharePoint functionality. Typical farm consists of web server running on IIS and SQL

database server. These can be deployed on a single computer or on two separated machines. Larger solutions usually consist of several Web and database servers.

### 3.4.3. SharePoint site structure

At the top of SharePoint site structure is a Web Application. It is a SharePoint's representation of IIS's Web Application pool. We can specify communication ports for our sites and authentication providers and techniques and database connections.

Second level is a Site collection. It is basically a regular site with some default requirements like permissions, group collections, administration settings and owner. It is usually used as a project main site. Any sub sites made in this site collection, inherits root site's settings. This inheritance can be break and you can create new sub sites similar to site collection within your project site structures.

Another level is a sub site. It is a site made by its parental site inheriting parental permissions, by default. A site nesting is unlimited and you can create as many sub site levels as your sane mind allows.

Penultimate level is a page. It is classic ASP.NET Web Page that shows information.

And last level is any item in the page. That could be list, document library, web part and anything you want.

### 3.4.4. Permissions

Not everyone has complete permissions to do whatever they would like to do. A security is absolutely essential especially for business deployment. Every user has to use his logon credentials – his user name and password. Only after proper authentication and authorization he can work with sites and their objects, based on rights, which the administrator has given him. It limits the user for concrete type of work. When higher rights are assigned, wider ranges of possibilities for accessing and making actions are granted.

At the top of SharePoint hierarchy is a Farm Administrator. He creates and manages all web applications, assigns rights and generally takes care about whole farm.

Second in line is Site administrator. He can be a project manager or someone assigned by the project manager. He cares about sites and working environment for project workers.

You can assign permissions to any object in SharePoint from SharePoint farms up to any single list.

Other people can have range different permissions for variety of tasks in the project, based on permissions for specified role definitions. By default, there are 5 role definitions – Full Control,

Contribute, Design, Read and Limited Access. You can create your own custom role definition by specifying role permissions.

Full Control is for administrators. It provides unlimited access for specified object (from SharePoint farm up to any web page item). Site owner has these administration rights.

Design permission level allows creating and editing web pages, lists, document libraries, style sheets, but cannot manipulate with items in lists.

Contribute permission level is assigned to site members. These users can add, edit and delete items in lists, but cannot manipulate with a site structure itself.

Read permission level is for visitors. They cannot manipulate data or site structure. They just can read information.

Limited Access permission level is assigned automatically by SharePoint and user (even administrator) cannot assign or change it. This permission level has been made for security reasons when user got an access to some object, but has no rights to access higher levels for the object. For example, if the user has rights to access a item in the list, but has not rights to access the list itself. Then SharePoint assign to the user a limited access for all higher level objects.

### 3.4.5. Web Parts

Web Parts are reusable customizable objects in SharePoint and one of the most important parts of SharePoint site design. Many sites - your collaboration portals are different but consist of the same "building blocks". Web Parts pack and bind together design and functionality into DLL, so SharePoint developers can use them again and they are not forced to start from the scratch.

Web Parts are basically ASP.NET controls implementing Web Part Control Set that enables to users to modify appearance and behavior directly from their browser and thus personalize them to user's needs. There are many types of Web Parts intended for various types of use and situations and developers can create their own for their special purposes. Some of them are visual and some are just functional.

Each web page can contains several Web Parts connected with databases, web services, methods, another Web Parts or just showing some information. They are usually easy to use even by nontechnical persons such as project managers. Programmer creates Web parts and any project worker can use them. Visual Web Parts usually have in-design settings, adjusting Web Part behavior and look.

**Web Part editing**

If you want to edit some web part attributes, open a page with a concrete web part, switch to edit mode, choose the web part you want to modify and new context tabs will appear on the upper panel. Then click on the Web parts tools tab – Options and choose Web Parts properties option from the menu. New dialog box will appears and you can in-design modify the Web Part – its look, behavior, layout and more. Various Web Parts have various options.

### 3.4.6. Web Page

Web Page is a main data storage consisting of text information and other data storage like libraries and lists. Web Pages include images, sub-pages, web parts, links and other navigation components.

SharePoint supports 2 elemental pages types. First is a main page of the root site, which is created as a wiki library page. With this page type you can add images, texts, links and web parts to the page anywhere you want without any restrictions.

Second type pages are based on elementary page template. This template consists of two parts and their settings are not possible to change - left one is a navigation panel and right one is a part for information. All others sub-site pages are like this type.

**Web page editing**

To edit page, click on the Page tab in upper ribbon panel and choose Edit option. Then a new ribbon context tabs opens in the upper panel with text formatting and tabs inserting.

A Format text tab is a ribbon very similar to the ribbon in Word for text font editing and options for paragraphs, text styles, layouts and so on. There is nothing really interesting here and everything is easily understandable.

A Second tab is an Insert tab. Here you can add new objects to your page. Excel tables, pictures, links, upload files, lists and Web Parts.

A New list button opens a dialog window with many most used common lists in three categories and possibility to create your custom. Choose the one you need for your page, enter a list name a click on OK.

When you click on the Web parts button a new dialog box shows up and you can choose a new Web part for your page in six categories. You can add calendar, tasks, links, XML viewer, image viewer and SilverLight (which is novelty in this SharePoint version) and more others.

### 3.4.7. Lists

Lists are key component of SharePoint for information storing. Most of them are stored just in lists. Even document library is kind of a list. If you want to add a new list to your site, you can create your own custom list or choose one of pre-defined templates.

All accesses to SharePoint database should be realized through lists. Direct approach to SharePoint database is possible although it is not recommended.

Lists are similar to database tables or excel tables, so columns represent data attributes and rows represent data records – one row for each record.

When the list is finished and some data are stored in it, you can design a new list view and define data sorting, grouping and filtering as well as setting styles, display limits and other.

**Columns**

To add a new column (attribute) to a list, click on Column button in List tab. There you can set properties for the column – column name and one of predefined attribute types like Single line of text, Multiple lines of text, Choice, Number, Currency, Date and Time, Lookup (to join the column with existing list), Yes/No option, hyperlink or picture, calculation based on other columns (ex.: calculation of final price based on base price and tax) and external data. When you have chosen type, you can describe a new column, set whether the attribute should be required or if has to be unique, validators and additional information unique for each type.

### 3.4.8. Document libraries

Document library is a main SharePoint component for storing and managing documents, creating custom metadata for them and custom views and version monitoring. Document library has an email alarm and can send email to administrator in a case of document change or another situation set by administrator. Document library contains basic templates for forms, image libraries etc.

### 3.4.9. Interface

SharePoint user interface was designed for simplicity and user friendliness. That is why the creators used well-proven web interface style known to anyone from web browsers. The interface has changed since last version using ribbons. Here I would like to describe main parts, features and navigation system.

**Upper panel**

Upper panel uses ribbon style known from Office family since 2007 version. For some users that can be controversial, but honestly, I really love that thing. If you like it or not you should know that

MS uses this ribbon style for interfaces more and more frequently and SharePoint is not an exception.

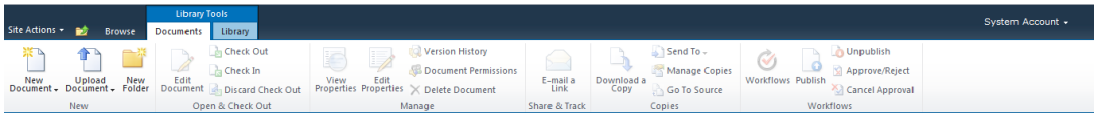


**Figure 4 - Upper panel**

### Site Action menu

There is a combo box Web Actions on the left side of the upper menu with different actions depending on your access rights. You can edit an actual page and add new content. Here is the place where you can create information storage, view all site content and new sub sites. Here you can access site permission settings and site settings.

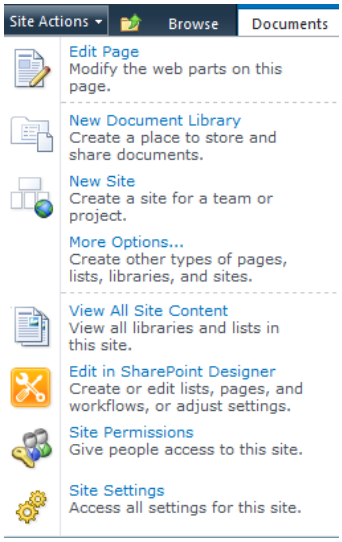


**Figure 5 - Site Actions combo box**

### Go up button

No matter how high rights a user has, here are no limitations in viewing all actual page content by clicking on a GO UP button. That comprises libraries, lists and sub-pages tied together with an actual page. The content is shown in hierarchy order as I described before and you can easily navigate from actual page up to the root.

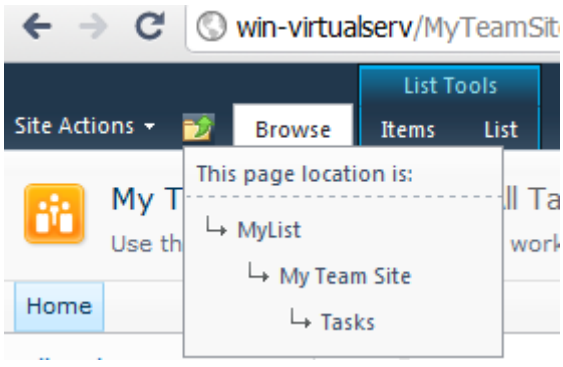


**Figure 6 - Page content view in hierarchy order**

### Page editing button and Context options

If you have site administrator rights, you can edit actual page by clicking on the edit button just next to GO UP button. Other tabs on the right side are various depending on actual page or object you choose. These tabs are showed only when a content of the page view or object related with tab

has sense. You can see a List Tool tab on the figure, which is one of the context tabs. In this case it shows options for a list.

### List Tools

As I said, lists are essential for any project and you will probably spend much time with List Tools tab. There you can deal with Items or the list itself.

In Item tab you can view, create, delete or edit any data records (rows) in your list directly, check workflows and attach an attachment.

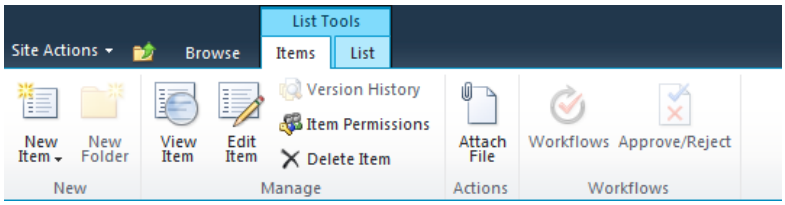


**Figure 7 - Item ribbon**

The List tab allows you to manage whole list – create and modify data views, export list to Excel, create Visio diagram, set permissions and workflow settings and set list settings.


**Figure 8 - List ribbon**

### List Settings

List settings contain all imaginable information and settings for the list like a name, Web address, title, navigation, all data views, validations, versioning and workflow settings and column settings. Here you can delete the list, create, delete or modify columns and manage everything except data (rows) in the list themself.

In SharePoint 2010 version is a novelty that you can specify how the SharePoint should react to column relations when deleted. You can choose Restricted delete option – that means that you cannot delete parental item in a case it has some children and thus break the relation. Second option is a Cascaded delete – all children will be deleted with their parents. Those items will be moved to recycle bin (that is a novelty in the SharePoint 2010). And if you change yur mind and restore parents from a bin, children will be restored with them and the relations won't be broke.

**Account menu**

On the right side of upper panel is an actual user's name. It is your name, with which you are logged on. Here you can edit your personal information like name, personal information, your picture, job title, contact information as well as local information like time zone, your language and if sites are prepared for different local settings, they can be adjusted and you can see sites in your native language. At the account setting page is even a link to set up your alerts. These can be added by you or by site administrator and can inform you about changes, comments, new posts or data. You can view and change account settings clicking on Account combo box.


**Figure 9 - Account combo box**

**Bookmarks**

Under the ribbon part is an area with site bookmarks. Objects in bookmark place are visible only when you have rights to access them. Bookmarks are editable and sharable.

### 3.4.10. SharePoint Central Administration

And now let's say few words about Central web Administration, which is important for farm and web application administrators and generally for people with higher permissions, who take care about SharePoint running.

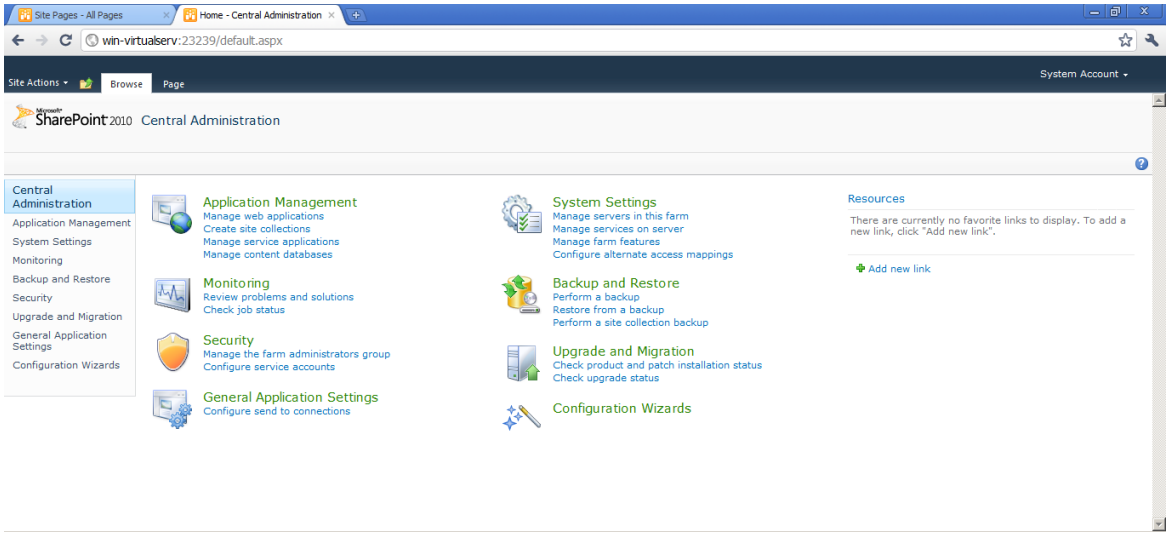


**Figure 10 - Central Administration**

We can get to Central administration through Windows Start by clicking SharePoint 2010 Central Administration element or browsing to Central Administration URL. Of course you have to have proper permissions to get there.

Here we can see many different settings options sorted to groups by type and logical functionality. Here you can manage all server settings, information, permissions, farm backing up, creating new Web Applications, monitor performance and issues, add new servers to the farm, enable and disable farm and web application features and more.

A complete list of function would be very long and most of them are not essential for SharePoint using and are a business of farm administrators.

Most important part is an Application management where you can create new site collections – new projects and manage them from the top view. You can for example configure alternate access mapping allowing accessing your project sites from multiple location (intranet, internet) and mapping several access URLs.

### 3.4.11. Site Settings

To get to Site settings click on the Site Actions menu and choose option Site Settings. Here are settings for administration actual site – adding new people and groups and assign them permission, manage galleries of site construction blocks - all parts our sites are made from like web parts, site columns, themes, solutions, templates and more. You have all these parts in one place.

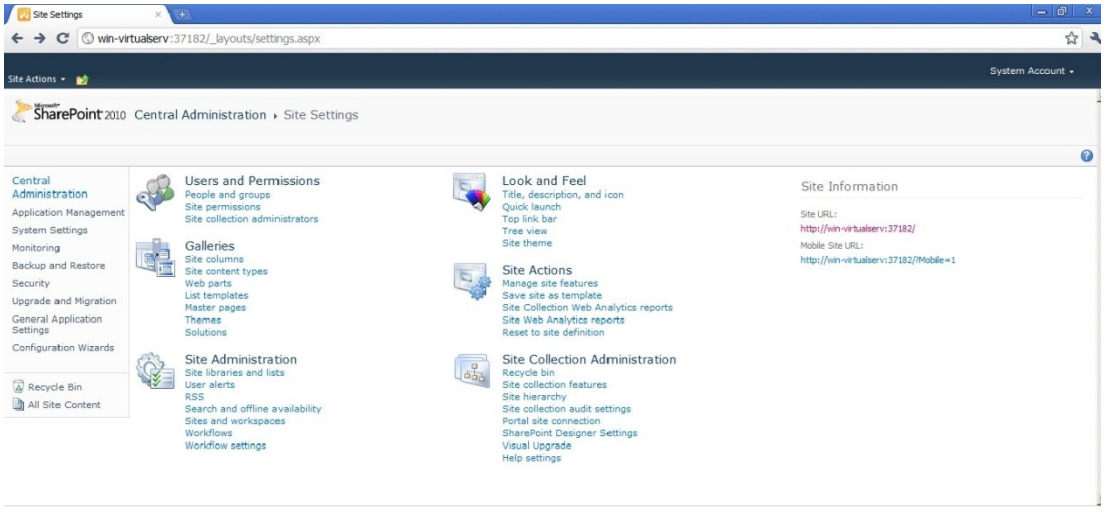


**Figure 11 - Site Settings**

For example Users and permissions category contains permissions administration actions and user accounts. The restriction settings are heritable within site collections and are applied to all sub sites of an actual web.

People and groups – here you can see and setup groups and users, who can access the SharePoint system. These groups and people are connected with users and group administration in Active Directory service. It can be a local server or server farm domain or just local machine. You cannot create new users/groups, but you can assign permissions to existing users in AD and even setup user/group information like email, phone, job position etc.

Site permissions – you can define which users and groups are capable to perform actions with actual site and its sub-sites.

Site collection Administrators – it is permission definition for whole site collection. Site collection administrators have a full control all over site collection.

# 4. How to create a new site in SharePoint environment

Now we know what SharePoint is and why it is very useful and we have some information about structure, objects, interface and settings and the time to use it just has come.

First of all we have to create a Web Application in Central Administration - Application management which is on the top of SharePoint hierarchy and serves for IIS identification.

Application pool includes one or more applications and we can define isolation rate between each Web application and their Application pools. Each of these Application pools works in separated process and thus errors of one application from one pool can't influence other applications in theirs pools. A good example is memory leak, which is limited right by the pool, because it has its own limited resources including memory and such an error cannot crash other applications or the server on which the defective application runs. In IIS7 is Application pool configuration much easier than in older versions because it is not necessary intricately set it manually anymore and Application pool is created automatically for any new Web Application. And subsequently is possible to change the settings [5].

When we create a new Web application, we can configure an authentication way. A name of a new Site on IIS, its port, path in file system for virtual directory, security settings, a name of public URL, basic information for Application pool, database information (a name of database server, a name of new database for our new Web application and authentication methods). We can even choose a mirror database server for better safety and administrator's calmer sleep and search server if a standard SharePoint Search server is not installed. Everything is configured by default, so in most cases you won't need to make extensive changes. Mostly all you need to change is IIS Web Site name and public URL.

Next thing in SharePoint hierarchy what we need to create is Site Collection, which is a root site of our Web Site. And again we create Site Collection in Application manager in Central Administration. Here we can specify a name of the Site, which will be shown as a title of every page that we create in the Site. Next we have to specify our site URL and we can choose a default Site collection root folder or any custom sub-folder.

For Site simplifying of creation we can choose one of a predefined most used templates arranged in several categories by the type of usage. It can be a blog, wiki page, Team Site, Document Workplace, publishing portal or just choose a blank page and start from a scratch. Template options differ depending on SharePoint version you use. Paid SharePoint Server version has considerable

more options. Then we have to choose primary and secondary administrator. Just one person can be used for each role and groups are not supported. Then click on OK button and site collection for your project is done.

Now you see root site for the site collection using template that you chose.

# 5. Implementation

Theoretical part is important for this diploma paper, but practical part shows what I have learnt and tells us how I can use this knowledge in practical way. The practical part consists of some basic information for the beginning – something you should know before you start. Then come implementation technologies and approaches that I have used during programming and issues I came across.

## 5.1. Before you start

### 5.1.1. SharePoint 2010 64 bit

SharePoint 2010 is 64 bit version and there is no 32 bit equivalent. When you access SharePoint by SPSite class, you have to compile the project to 64 bits, because SPSite uses some unmanaged code and thus cannot work with different architecture. Keep it on mind when preparing your project.

### 5.1.2. SharePoint 2010 .Net Framework

SharePoint 2010 uses .Net Framework 3.5 and is NOT compatible with version 4.

### 5.1.3. SharePoint and remote server

When the server on which you want to access is remote – that means it is not a part of your internal (school, company) network – you cannot communicate, read and send data by classical SharePoint API – SPSite, SPWeb, SPList way etc. That's because SharePoint won't allow you to establish connection from outside; for safety reasons. If you want it, you have to do it via SharePoint Web Services.

Easiest way is to add service references to you project and use directly site structures. Disadvantage of this solution is that you have to know the structure during programming and you cannot use the same application for any random site.

## 5.2. Implementation technologies and approaches

I have decided to use my laptop with Windows 7 operation system as a development machine; for several reasons. SharePoint 2010 supports installation on client operation systems – just with a few modifications. Otherwise I would be forced to use virtualized Windows 2008 R2 version, which is important for deployment testing, but highly reduce performance of my notebook and makes development slow and uncomfortable. It is mainly due to RAM size limitations. My notebook has only 4GB RAM for both host and guest machine and operation systems and Visual Studio, SharePoint, SQL databases are quite demanding applications.

### 5.2.1. Projects' general communication architecture

None of my application communicates directly with SharePoint. As I said it is because SharePoint doesn't allow direct access from remote computers. If you try to open connection, the exception is thrown.

This problem has 3 potential solutions. First, don't try it. But that could be limiting. My test virtual server was on remote computer and I was out of domain. Secondly, use SharePoint internal web services, but the client application would have to be written specifically for the particular site. Third, write you own custom web service or windows service. A web service would have to be installed to target SharePoint server whereas Windows service on some computer in the SharePoint's internal network and communicate via ASP.NET services or WCF services.

I have chosen Windows service option with WCF services.

So, both my applications communicate with my custom server-side Windows service via Windows Communication Foundation and this service sends all requests to the SharePoint server. Why I chose WCF instead of classical ASP.NET Web Services is described in the next chapter.

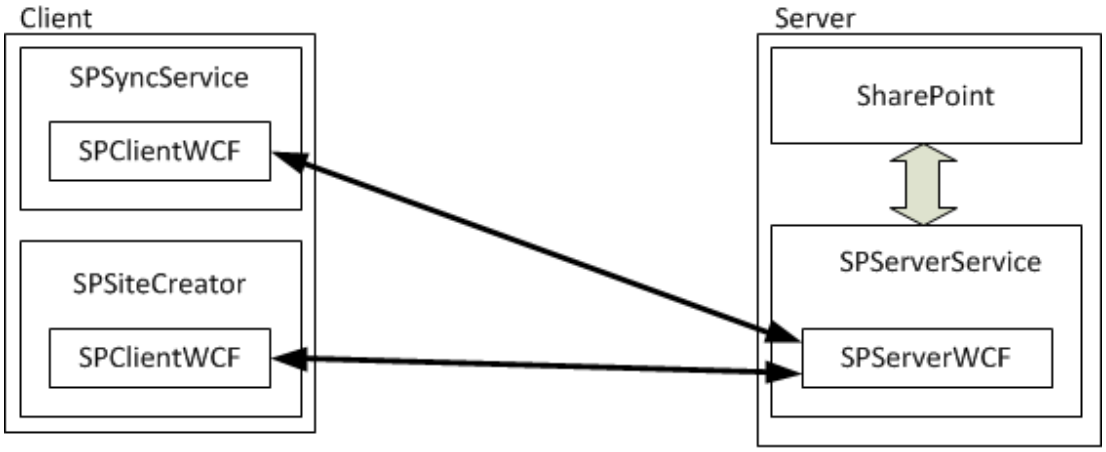


**Figure 12 - General communication architecture**

### 5.2.2. Windows Communication Foundation

I have decided to use WCF services instead of classical ASP.NET Web Services. Those are more flexible, faster and they are a part of much larger technology and I have used the opportunity to learn some basics.

If you want to use WCF as well, check if the WCF Activation feature is enabled on your server in Control Panel – Programs and features – Windows Features – Microsoft .NET. If it is not, turn it on.

Under normal circumstances everything should work normally, but if you have any problems with WCF on IIS, try this procedure:

> Run Command Prompt as administrator, go to "c:\Windows\Microsoft.NET\Framework\v3.0\Windows Communication Foundation" and register WCF service model executing "ServiceModelReg –i" command. ServiceModelReg register WCF and WF on the machine.
>
> Now run Visual Studio Command Prompt as administrator, go again to "c:\Windows\Microsoft.NET\Framework\v3.0\Windows Communication Foundation" and execute "aspnet_regiis.exe –i" for scriptmaps to ASP.NET ISAPI version.

When developing, don't forget to target the .NET Framework to version 3.5!

A semantic of ASP.NET request processing fundamentally differs from WCF. So it is possible that you will need to turn on ASP.NET compatibility mode to ensure WCF compatibility with ASP.NET.

To do so, add an AspNetCompatibilityRequirementsAttribute set to Allowed or Required in front your service class definition.

```
[AspNetCompatibilityRequirements(RequirementsMode =
AspNetCompatibilityRequirementsMode.Allowed)]
    public class Service1 : IService
    {
       …
    }
```

### 5.2.3. Web Parts developing

When you install SharePoint to your computer, a set of SharePoint tools is installed to your visual studio too. Both these technologies are designed to cooperate together. With Visual Studio 2010 comes with much larger support for SharePoint comparing to older versions. You don't have to edit web.config files anymore and upload your DLLs to Global assembly cache. I checked both deployment ways and I have to say that this really speeds up development.

To start create web part, open new project, expand SharePoint tab and choose Visual Web Part template, type project name and press OK. Now you should synchronize your project with some Web site. Verify, if it works by pressing validation button and choose one of two deployment solutions. First, you can synchronize it with real farm. Second, you can use sandbox solution.

Your project is ready and you can start to program it. You can deploy it easily pressing F5 or click with right mouse button on your project and choose Deploy.

**Manual web part deployment**

If you can't synchronize Visual Studio with your site or you just want to deploy some purchased web part or web part from internet, which is distributed as DLL, you have to do it manually. Take the DDL file and upload it to GAC (global assembly cache) or to bin folder in the site's root virtual directory. Library in virtual directory could cause some trust issues in some cases, so upload the library to GAC and you won't have any problem.

Then you have to inform SharePoint about this library. To do so edit site's web.config file in site's root virtual directory, find SafeControls tag and register new assembly.

```
<SafeControls>
    <SafeControl Assembly="BackupViewer"
Namespace="BackupViewer" TypeName="*" Safe="True" />
…
</SafeControls>
```

The assembly attribute defines a library for the web part. Then go to Site Settings in SharePoint and click on Web Parts in Gallery category. Click on New Document and find and put a check on your web part in new dialog box. Now your site knows about your web part, which is registered as a new site's feature. Now go to Site Settings again and click on "Manage site features" link and activate your web part and it is done.

### 5.2.4. Windows Service Programming

Windows services are long-running applications that run in the background in their own Windows Sessions and are ideal for server applications or applications that can run without user interference. Services can run even in security context, which is different from the current logged user's context and are started before the user logs on [6].

Windows service is a special type of windows application with unique implementing and deploying ways and application functions. Anyway, you can program windows service in regular Visual studio. Visual studio has a predefined Windows Service template that easing programming starts, but not all Visual Studio versions includes this project template. If your version is the limited one, there are no functional limitations – you just have to choose for example a Console project template and modify it by yourself.

At minimum your application has to inherit from ServiceBase class and your main method has to call the Run method, which loads the service to Service Control Manager and you must override OnStart and OnStop methods.[5] You have to take note, that you cannot just run the executable file to run the service. First you have to create installation component to install it to a computer.

An interaction of windows service with a user is very difficult to implement, so services usually don't interact at all. Mostly it is not even necessary, because services often perform simple and repetitive work. Service can exist in one of three states: Running, Paused and Stopped. A service can automatically switch to one of the states or when user intervenes.

A Windows Service development is more tricky compare to developing other Visual Studio application types, because you cannot test it and debug it so easily by pressing F5. This procedure doesn't work. First you have to install the service to your computer, start it and then attach debugger to the process in which it is running and then use a breakpoint to debug the code. And even then you cannot debug all parts of – a debugger can be attach only to currently running service, so OnStart method cannot be tested.

That is why I chose a different approach. I created Windows Service Emulator as a regular Console application and tested whole code in it. When the testing was done, I have changed the code from Console Application to Windows Service. Development was much easier.

**Windows service deployment**

When your service is ready to deploy, you have to add ServiceProcessInstaller and ServiceInstaller classes. In ServiceInstaller properties set a name of your service and start type (Manual, Automatic or Disabled). ServiceProcessInstaller allows you to decide how an Account the service is logged like:

- User: service uses current user's credentials and works in the context of the user's account
- Local Service: service works in the context of non-privileges user on the local computer and presents as anonymous to any remote server.
- Network Service: service works in the context of non-privileges user on the local computer and presents computer's credentials to any remote server.
- Local System: service works in the context of extensive local privileges and presents computer's credentials to any remote server.

My service works in user context.

[5] How to write Windows Service programmatically: http://msdn.microsoft.com/en-us/library/76477d2t.aspx

Now you have to add Visual Studio Installer. Add new project to your solution and choose Setup Project template. Now you have to add service output to Setup Project (the installer). To do so choose one of the output groups (primary output by default) and don't forget to choose your target service project in combo box. Now your project's output is added to the installer. Build and install it.

**Service installing**

I have implemented automatic service installation to Service Control manager just after the service installation. That radically saves my time during service testing period and simplifies deploying for end-users. To do this I had created a new Custom action - Installation project template and overrode Install and Commit methods.

Install method prepares installation and sets variables. To get a program target location chosen by a user I have used an IDictionary variable stateSaver and defined a new key-value pair.

```
public override void Install(IDictionary stateSaver)
        {
            base.Install(stateSaver);
            stateSaver.Add("TargetDir",
text.Parameters["TARGETDIR"].ToString());
        }
```

Then I set the CustomDataAction property in Visual Studio Installer's Custom Action view.

```
/TARGETDIR="[TARGETDIR]\"
```

The first TARGETDIR is my value, which I defined in Install method and the second TARGETDIR is Visual Studio Installer's internal property in Install method. Finally I have used a directory that I got from an IDictionary savedstate variable in overridden Commit method and I could install the service automatically using installUtil.

.

```
public override void Commit(IDictionary savedState)
        {
            base.Commit(savedState);

            ProcessStartInfo info = new ProcessStartInfo();
            info.FileName = installUtil;
            info.Arguments = "\"" + savedState["TargetDir"] +
"\\SPSyncService.exe\"";
            info.Verb = "runas";
            Process.Start(info);
        }
```

Now every time I install the program, the service is automatically registered to a computer and is ready to use.

## 5.3. Implementation issues

At this part of my diploma I would like to describe problems I had to face during implementation and ways how I have solved them. Some of the problems could be specific for my hardware and technical configuration. Other could be more widely spread among SharePoint users.

### 5.3.1. SharePoint installation on Windows 7 client OS

1) Modify SharePoint installation and allow installing on client operating systems [7]
   - Execute installation exe file in Command prompt with extension "/extract:c:\SharePointFiles"
   - Go to "c:\SharePointFiles\Files\Setup"
   - Add "<Setting Id="AllowWindowsClientInstall" Value="True"/>" to <configuration> tag in config.xml configuration file
2) Install prerequisites from "c:\SharePointFiles\PrerequisiteInstallerFiles\FilterPack\FilterPack.msi"
3) SharePoint needs enabled IIS features
   - You can do it manually in Control Panel – Programs and features – Turn Windows Features on or off or execute in command prompt:

```
start /w pkgmgr /iu:IIS-WebServerRole;IIS-WebServer;IIS-CommonHttpFeat
ures;IIS-StaticContent;IIS-DefaultDocument;IIS-DirectoryBrowsing;IIS-H
ttpErrors;IIS-ApplicationDevelopment;IIS-ASPNET;IIS-NetFxExtensibility
;IIS-ISAPIExtensions;IIS-ISAPIFilter;IIS-HealthAndDiagnostics;IIS-Http
Logging;IIS-LoggingLibraries;IIS-RequestMonitor;IIS-HttpTracing;IIS-Cu
stomLogging;IIS-ManagementScriptingTools;IIS-Security;IIS-BasicAuthent
ication;IIS-WindowsAuthentication;IIS-DigestAuthentication;IIS-Request
Filtering;IIS-Performance;IIS-HttpCompressionStatic;IIS-HttpCompressio
nDynamic;IIS-WebServerManagementTools;IIS-ManagementConsole;IIS-IIS6Ma
nagementCompatibility;IIS-Metabase;IIS-WMICompatibility;WAS-WindowsAct
ivationService;WAS-ProcessModel;WAS-NetFxEnvironment;WAS-Configuration
API;WCF-HTTP-Activation;WCF-NonHTTP-Activation
```

4) Install SharePoint
5) Configure SharePoint Products Configuration Wizard.[6]
6) Go to SharePoint Central Administration with your actual user account logon credentials

### 5.3.2. Security problem – cannot modify/create Web applications on client OS

If you cannot create new web applications or modify existing ones because NEW, EXTEND and DELETE icons in Web Application managements are grey (disabled), you have a security problem between your browser and SharePoint server. In my case I couldn't fully use SharePoint with

[6] - If you have any problems during configuration on Windows Vista/7 check this link for possible solution: http://myspexp.com/2010/05/31/configuration-failed-when-installing-sp-2010-on-windows-7-failed-big-time-3/#comment-55

Google Chrome browser and I didn't find a way how to go around the problem. No matter if It ran it under Administrator privileges or not.

To solve the problem you have to put up with Internet Explorer or be luckier than I was. So, how to solve the problem? First go to Control panel and click on Internet Options. Go to Security tab, switch to Local Intranet and click on Site button. Check if your "Automatically detect intranet network" checkbox is checked. Then click on Advanced button and add the site url to intranet zone. Even now you have to access to Central Administration with Internet Explorer running as Administrator.

### 5.3.3. SharePoint is accessible, but your credentials are not accepted

Can you access SharePoint site and your credentials are required, but no matter how many times you type them, they are not accepted? Problem is not with your password or username. Problem is in SharePoint/IIS authentication. SharePoint demands Windows authentication, by default. If you have aggressive firewall or you need to access the server via VPN, windows authentication may not be possible. The trick is in the "by default". You need to enable basic authentication and then all you need is your login name and password.

But be careful. If your site doesn't support SSL protocol, your credentials are sent as a plain text.

And probably you don't want to disable windows authentication for all users. So, first extend your site in SharePoint Central Administration for new port. This will add another application pool for your site in IIS, and then change authentication settings in Extend configuration dialog box. Now you should have an access to SharePoint.

### 5.3.4. Access to SharePoint from Internet when on virtual machine

When you operate your server on a virtual machine as I do, is important to check your network settings of your virtualized server for accessibility from the net. Some virtualizing applications use NAT as a default network setting for virtual machines. But NAT (Network address translation) could work as a firewall in some cases and your server may not be accessible from outside of NAT. If it happens you just have to change your network settings to Bridge, so your virtual machine will share your internet connection with your host computer. It is the easiest way.

Assure that you have permissions to access your server from your provider and he does not block the communication with server.

### 5.3.5. Access to SharePoint when you are behind a home router

When using wireless or other router at home, you have to forward requests to your internal IP address to get a server response. Then you can access your SharePoint from Internet.

That is not all. My test Site uses HTTP port 80 and router forwarding settings didn't allowed to forward requests to this port. What to do now? Easiest way is to extend access possibilities to another port by mapping alternative access and then forward to the new port. If it still doesn't work, set bindings in IIS manager. Or extend the site access in Web manager. This is similar to a alternative mapping, but this will create new IIS application pool for the same SharePoint site and you can set differ authentication settings and differentiate authentication procedures to each access way and spread the load.

### 5.3.6. Testing problem

How to test your SharePoint application? Unit tests are usually good way how to do it. However combination of SharePoint development and unit tests in Visual Studio 2010 is one big problem from the beginning.

Visual Studio is 32bit application due to 32-to-64 migration costs and performance difficulties (developers' declarations [8]) and SharePoint 2010 requires 64bit operating system. But Unit Tests use some native COM objects that can't mix 32 and 64 bit DLLs. So when you try to connect SharePoint using SPSite class, the System.IO.FileNotFound exception is invoked because of communication problems. This is not only problem of Unit tests but all applications built for 32 bit platform and that can be solved with changing Building -> … in project properties.

But this is not the only problem. If you try to debug your code now, another exception shows up, telling you that Microsoft SharePoint does not support version 4 of the Microsoft .Net Runtime.

Normally you would change project framework version from .NET 4 to .NET 3.5 in project properties, but that would be too easy. Visual Studio won't allow you to change it or even to create a new project with this setting. Visual Studio forces you to run your unit tests on .NET 4 framework, which is not supported by SharePoint. That is not a bug. It is made this way by design. Luckily I was not the only one who had problem with this design concept and more people were quite upset with that, so Visual Studio developers reconsidered their decision and allow to change framework version with Service Pack 1.

In my case Visual Studio didn't update automatically via Windows Update and I was really surprised that I didn't have an current version. If you have the same problem, you can download Service Pack 1 from Microsoft sites[7]

Updating takes about an hour.

[7] http://www.microsoft.com/downloads/en/details.aspx?FamilyID=75568aa6-8107-475d-948a-ef22627e57a5

Any other project has to be compiled for 64 bit platform and 3.5 .NET framework as well. But then a setup change is usually without problem.

### 5.3.7. Windows Installer setup

If your project includes some 64 bit libraries – doesn't matter if it is one of references or a target platform of your compiled applications, you have to change Installer's Target Platform property from x86 to x64, because 32 bit installer cannot pack 64 bit DLLs and the installer project wouldn't be possible to compile.

If you want to use Custom Action to customize the installation process, the Custom Action has to be compiled in 32 bits because of presence of internal non-managed code. But it is not a problem, because 64 bit Windows installer can cooperate with 32 bit Custom Actions without any problem.

# 6. Projects

I have developed 4 applications and one web part in 3 projects for this thesis. All of them are somehow connected and altogether form one coherent project. SharePoint Site Creator (SPSiteCreator) creates new Site collection, then SharePoint File Synchronizer (SPSyncService and SPSyncUI) synchronize desktop folders with SharePoint libraries and back up files. All three applications in both projects share communication with SharePoint via Windows Communication Foundation and SPServerService. You can find more information about communication architecture in chapter 5.2.1. – Projects' general communication architecture. The last project is a web part illustrative example BackupViewer showing file backups and can manipulate with them.

## 6.1. SharePoint Site Creator

If you have to create many new project sites for students, traditional way to do it is through SharePoint Central Administration. That may be quite time-demanding. And there is an opportunity for my program. You can easily create a new Site Collection in SharePoint structures with it.

The application communicates with server via WCF, and the server-side windows service does all the necessary accesses to SharePoint.

### 6.1.1. User guide

**Distribution**

The application is distributed as Setup.exe and SPSiteCreator.msi installation files in the SPSiteCreator folder.

The server-side windows service is distributed as Setup.exe and SPServerService.msi installation files in the SPServerService folder.

**Deployment**

Run Setup.exe file. Choose your Program location in Install Wizard and install the program.

Install SPServerService application on the server running Setup.exe from the application folder SPServerService. At the end of the installation, the service is registered to Service Control Manager and you are asked for your credentials. You have to have administration rights or contribution rights for the web site in which you want to upload.

**Windows Communication Foundation service configuration**

If you are not satisfied with default service endpoints, for a reason that the port 8888, which the application uses is occupied with another application or for reasons of port forwarding or firewall settings or anything else, you can change them easily without need to build the application again.

Go to the client application folder (C:\Program Files\SPSiteCreator, by default) and edit SPSiteCreator.config file. Change the endpoint where the server-side service listens by changing address attribute.

```
<endpoint address="http://localhost:8888/SPServerWCF/Service"
          binding="basicHttpBinding"
          contract="WCFConnector.ISPServerWCF"
name="WSHttpBinding_ISPServerWCF" />
```

Of course you have to do the same in the SPServerService.config file in the SPServerService folder on the server and edit baseAddress attribute with the same address.

```
<host>
    <baseAddresses>
      <add baseAddress="http://localhost:8888/SPServerWCF/Service" />
    </baseAddresses>
  </host>
  <endpoint address="" binding="basicHttpBinding"
contract="SPServerWCFProvider.ISPServerWCF" />
```

And then restart the service on the server.

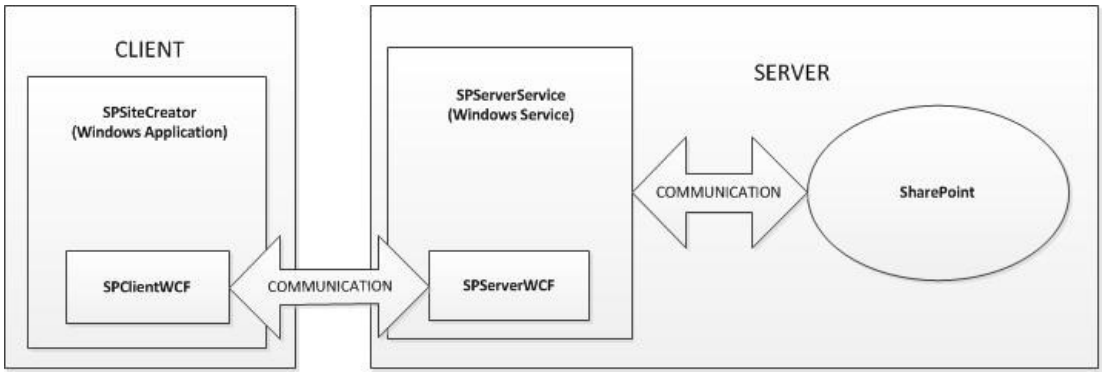


**Figure 13 - SPSiteCreator Application Scheme**

**GUI**

To create new site collection for your project, run the application, type and URL of a parent site for your projects (like http://myServer) and validate it. Validation is important – the program will checks whether the server-side windows service has all rights necessary for access to the parental site and checks a connection to the server. If the parental site and an access to the server are validated, you will see an icon of a green check sign and new site collection controls appears.

Enter a new site title and an address based on parental site, description and choose one of supported default templates for the site. At the moment that you choose a template, you can read its short description for it on the right bottom part of the screen.

If you want, you can easily add some document libraries and site owners or contributors for the site by clicking on specific button. When adding new site users, write their logins to the textbox and separate them with a comma. The textbox looks like a People Picker in SharePoint and its behavior is similar. After validation correctly validated logins are underlined and failed validations are stroke. Then choose their permission level. Users can have Full control rights for administration, contribution rights for site members or "read only" rights for visitors. Libraries and site members are optional and you can add them later in Site administration, but you have to assign a site owner – somebody with administrator rights. Those rights are not inherited from parental site. Permissions for a new site collection are unique and somebody must have rights to manage new site collection. Owner's email and name are optional too. If you won't type owner's name, the login will be used instead.

Be patient, parental site and user validation takes some time. Now click on "Create Site" button, wait about 10 seconds and everything is done.

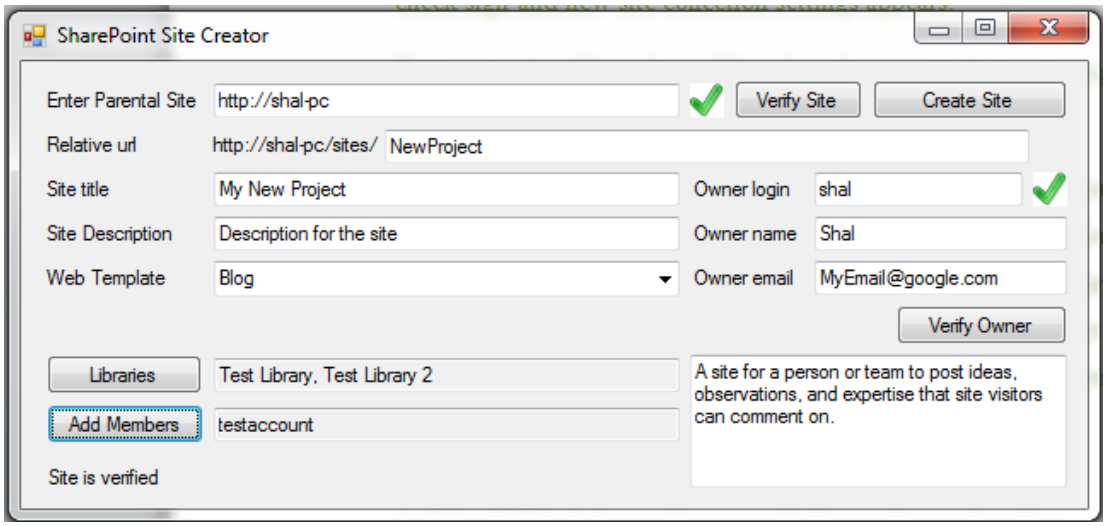


**Figure 14 - SPSiteCreator GUI**

### 6.1.2. Implementation

At the beginning, I had to clarify, which part of SharePoint structure I want to create. Generally, you can create new Web Application, Site Collection or Web Site. Each of these options has specific requirements and is made for different purposes. Basically, you can create new project site using any of this options. Web Application manages security and access way. It is SharePoint's personification of IIS Application pool. Site Collection is a root site and Web Site is a sub site of its root site inheriting site permissions.

I have chosen a Site Collection structure level for new projects.

Site members and site owner has to exist on the server machine or domain in which the server belongs. At first the application checks whether the user exists in Active Directory for the domain. If the computer is not in domain, it checks local machine. If user exists, program ensures the user on the server. That means, that the application checks whether the user already exists in a SharePoint server database or whether he can be added to the database. If everything is ok, the users name is underlined in the textbox.

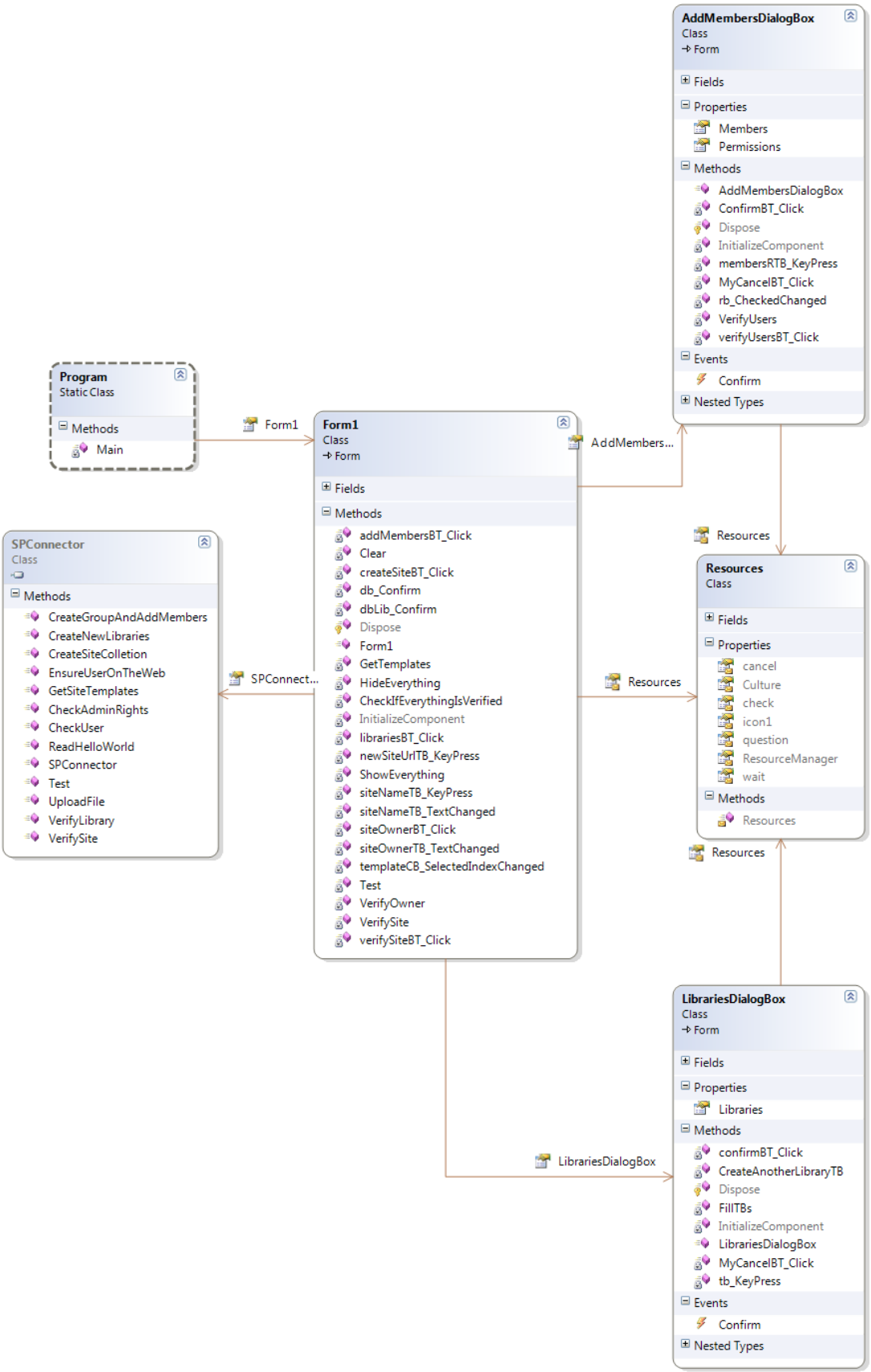


**Figure 15 - SPSiteCreator Class Diagram**

### 6.1.3. Project conclusion

I hope that my application helps to increase SharePoint usability in school, because the SharePoint. Strong side of SharePoint – the managing in web browsers, can be sometimes very annoying and if you have to go through all the administration to complete simple tasks, your motivation could decrease quite rapidly. With my program it takes only a minute to create a new project site.

## 6.2. SharePoint File Synchronizer

My second application uses Windows Service functionality for long-run on the background, monitors specified folder or folders and in the case of that a new file or files was copied to the folder or some file in the folder was changed somehow, it connects to SharePoint's particular library via server-side windows service and uploads all the files there. The service supports unlimited number of registered libraries and unlimited synchronized (monitored) folders for each of the library. Each library is independent of any other registered library.

The list of libraries and folders is stored in SPSyncService.xml configuration file, which is stored in Program Files\SPSyncService on the local machine.

Second application for this project is GUI made for easy interaction-like behavior of the service with user without necessity of xml configuration file editing. It is called SPSyncUI.exe and it is easy-to-manage application supporting checking for folders and libraries duplicities and dependencies.

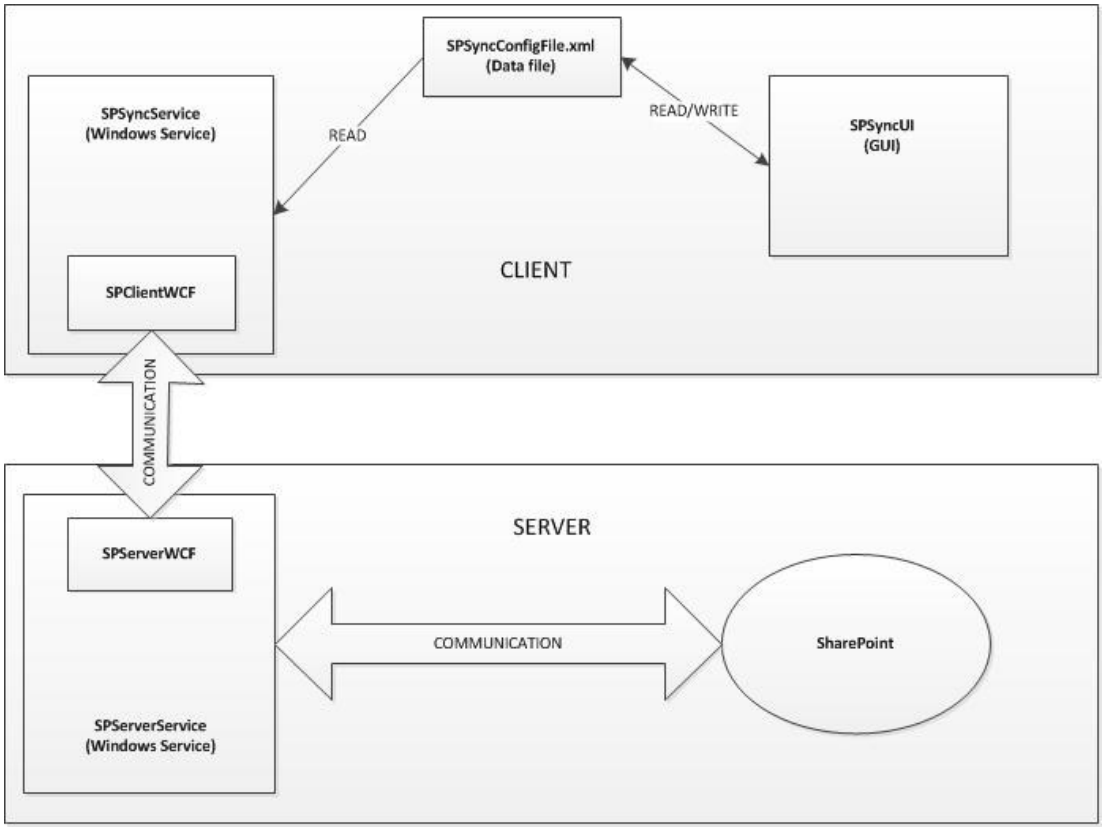


**Figure 16 - SPSyncService application scheme**

### 6.2.1. User guide

**Distribution**

The application is distributed as Setup.exe and SPSyncService.msi installation files in the SPSyncService folder.

The server-side windows service as distributed in Setup.exe and SPServerService.msi installation files in the SPServerService fodler.

**Deployment**

Run Setup.exe file. Choose Program location in Install Wizard and install the program. A service configuration file will be stored in %Program Files%\SPSyncService, no matter the location you chose. That is because of need for exact location of this file to avoid storing in the registry. At the end of the installation, the service is registered to Service Control Manager and you are asked for your credentials to run the service in your user context to get privileges to upload files.

Install SPServerService application on the server running Setup.exe from the application folder SPServerService. At the end of the installation, the service is registered to Service Control Manager

and you are asked for your credentials. You have to have administration rights or contribution rights for the web site in which you want to upload.

**Windows Communication Foundation service configuration**

If you are not satisfied with default service endpoints, for a reason that the port 8888, which the application uses is occupied with another application or for reasons of port forwarding or firewall settings or anything else, you can change them easily without need to build the application again.

Go to the client application folder (C:\Program Files\SPSyncService on default) and edit SPSyncService.config file. Change the endpoint where the server-side service listens by changing address attribute

```
<endpoint address="http://localhost:8888/SPServerWCF/Service"
            binding="basicHttpBinding"
            contract="WCFConnector.ISPServerWCF"
name="WSHttpBinding_ISPServerWCF" />
```

Of course you have to do the same in the SPServerService,config in the SPServerService folder on the server and edit baseAddress attribute with the same address.

```
<host>
      <baseAddresses>
        <add baseAddress="http://localhost:8888/SPServerWCF/Service" />
      </baseAddresses>
    </host>
    <endpoint address="" binding="basicHttpBinding"
contract="SPServerWCFProvider.ISPServerWCF" />
```

And then restart both services.

**GUI**

Application consists of 3 main parts – biggest panel on the left of the screen with two buttons above it. Let's call it Synchronization panel. On the right side of the screen are two panels. Upper one shows you whole configuration structure in collapsible tree view. Bottom one is a textbox for program information output that tells you what happened.

The synchronization panel shows you all registered libraries with their folders. Each of the libraries has its own panel with controls to interact with configurations for the specific library.

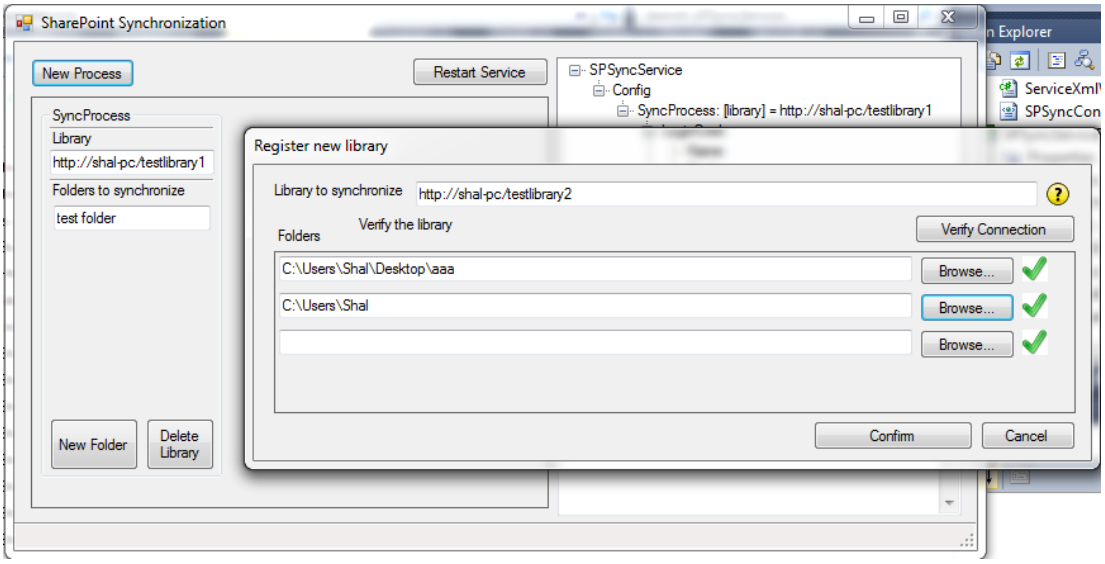


**Figure 17 - SPSync GUI**

## Configuring

Run Application GUI SPSyncService.exe in the installation folder.

*! The service needs to be restarted after changes. You can do it manually in Service Controller or just press Restart Service button in GUI application. !*

Register new library

Click on New Process button and a "Register new library" will show.

Type the library you want to register and validate it. Validation is important to ensure that the library really exists and you have permissions to access it and the library is unique. The library address is not validated immediately because of the time the validation needs.

Enter one or more folders by typing or choose it in Open folder dialog. The folders are validated immediately for existing and uniqueness. Duplications are prohibited.

If the folder or site is validated, a green check symbol is shown. If validation failed for some reason, a red cross is shown. The reason of validating failure if written in notification label below textbox for library addresses. A questing mark icon is shown if library validating is required. That happens if you change the address.

If all icons are green, click on the Confirm button to change configuration file.

Change registered library

Click on library textbox on the main screen. New dialog box appears and you can change the library address. The library has to be validated.

Delete registered library

Press Delete library button and confirm.

Add new folder

Press New Folder button, type the folder path in the dialog box or choose in Open Folder Dialog box. The folder is validated. Then confirm.

Edit or Delete folder

Choose folder you want to delete or edit, click on the textbox and in the dialog box which appears edit it and confirm or press the delete button.

### 6.2.2. Implementation

It was obvious from the beginning that the application will be developed as a Windows Service – long-running application running in the background. It could periodically checks monitored folders for changes, but implementing such functionality would be difficult to manage and synchronize with other library synchronizations. So I have used the functionality of FileSystemWatcher class made especially for these purposes. You just have to add a monitored folder and in the case of some changes had happened in the folder, the event arise. You can specify what folder events should be monitored – creating new files, renaming, changing, deleting.

I have implemented event handlers for creating, renaming and changing events. Deleting is not implemented, because deleting file from the monitored folder doesn't necessarily means that user wants to delete the file even from the library.

But work with FileSystemWatcher is not easy. And I had to face several bug-like situations. Those problems are not quite bugs. It is more likely error by design. The design of the class behavior could be useful in very limited situations, but mostly causes many problems and there are not any logical equivalents.

First of these so-called bugs arises!!!! events immediately when something happens. For example – when you copy a file to the folder, the event arise when copying begins and the file is not there completely and thus the file is locked and you cannot do anything with it. You cannot even easily check if the file is locked or not (if there are some handlers in the system for the file). And FileSystemWatcher doesn't support anything like OnCreatedWhenFinished event. Mostly used

solution for this problem is to manipulate with the file, expect an exception, put the thread to sleep for a while and manipulate with it again in a loop. It is not the most elegant way, but it works.

Another problem with FileSystemWatcher is multiple event rising. Some applications like Notepad saves files in parts. That arise events everytime the part is saved (the file is changed). In some cases OnCreated event arises multiple times when you copy several files to the monitored folder. I didn't find out why, but it probably has something to do with buffered file copying. Anyway, buffered or not, it is very annoying. Even more annoying is that the events are not raised in logical order. If you copy for example File1.txt, File2.txt and File3.txt, the order of events could be something like: File1.txt, File1.txt, File2.txt, File3.txt, File1.txt, File3.txt. I have solved this problem by creating event register that stores filenames and times of event risings. If the filename of the file which raised the event is the same like another record in the register in specific time span, the program ignores it. If the filename is new or the record time span is longer, event is processed and new record in registry is created. Every time the registry check is performed, the cleaning procedure is executed and records older than another defined time span are deleted. That prevents record cumulating.

I have put a file filtering in DirectoryWatcher (that is my managed version of FileSystemWatcher) checking what file was created in the directory. If the file ends with "tmp" suffix or starts with "~$" string, that means those files are just temporary files made by Office applications during editing. Those files could be created dozen times and sending them is not only impropriate, but even inadmissible.

When the files we want to upload to a server's library are identified, the folder is searched in the list of my custom classes LibraryWithFolders. When a library with registered folder is found, library path, file path and filename are sent to SPConnector. That happens with every match in the LibraryWithFolders list.

SPConnector reads the file, makes an array of bytes of it and sends them with all necessary information to the server-side service. The file is reconstructed on the server and uploaded to the library.

When the file gets to other side and is reconstructed, the server-side windows service checks, if the file already exists on SharePoint and if it does, service takes the file on the server, makes a backup and uploads a new version to specified library.

Backup creating is complex process. At first the service checks, if backups are supported on the site. It checks backup directory and settings file. If they don't exist, it creates the directory and a default settings file.

Backuping keeps file location. For example: If you have a file "MyFile.txt" at location "MyDirectory/MyFile.txt", the backup location is "backups/MyDirectory/MyFile.txt.backup1". The mask for backups is ".backup" plus backup number. Higher number means newer backup.

Number of backups is not programmatically limited and is defined in backup settings file "backupSettings.xml" in backup root directory "backups".

When number of defined backups is depleted, an oldest backup is deleted and the rest of backups are renamed, so the newest one can be stored with highest number.

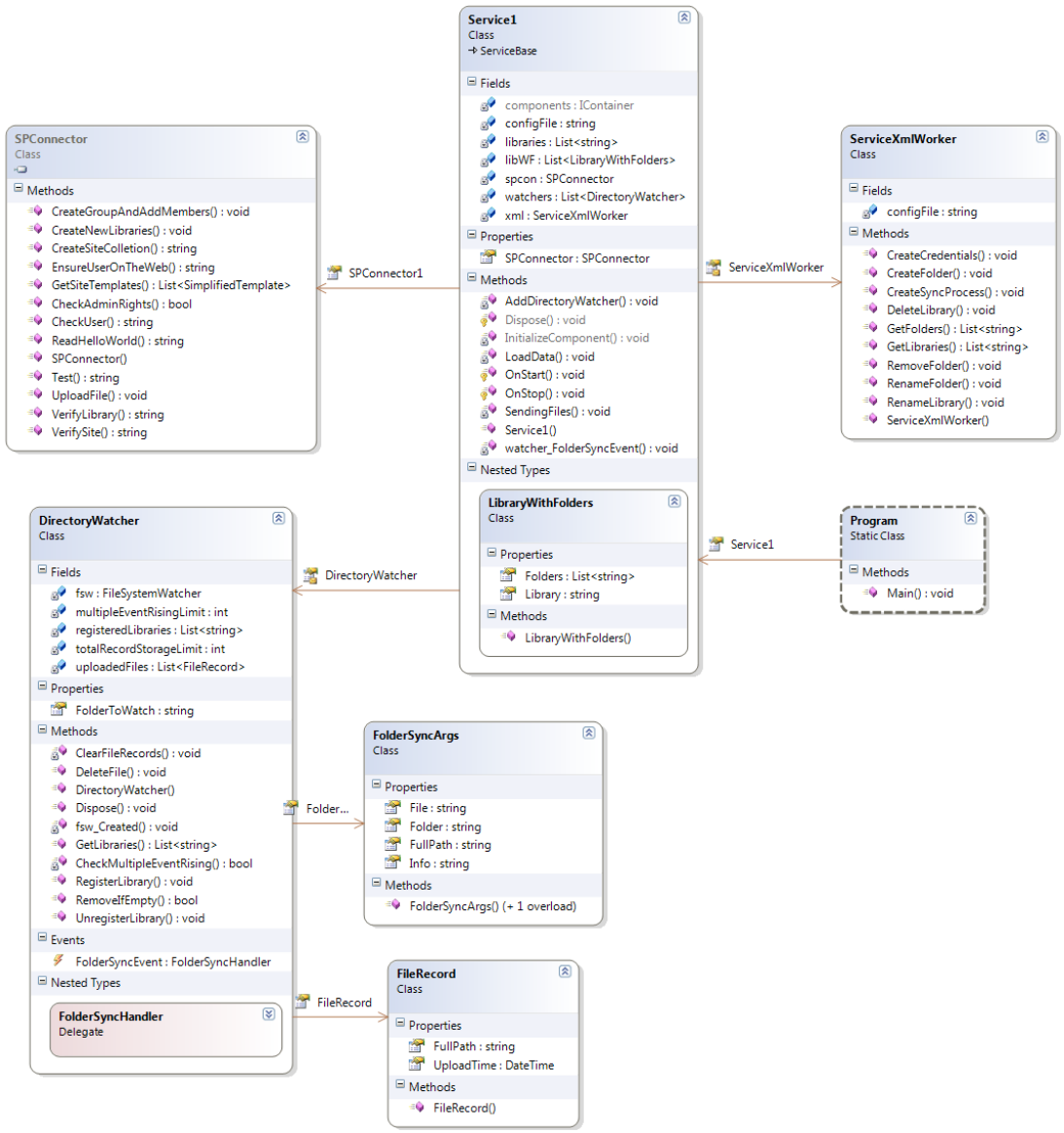


**Figure 18 - SPSyncService Class Diagram**

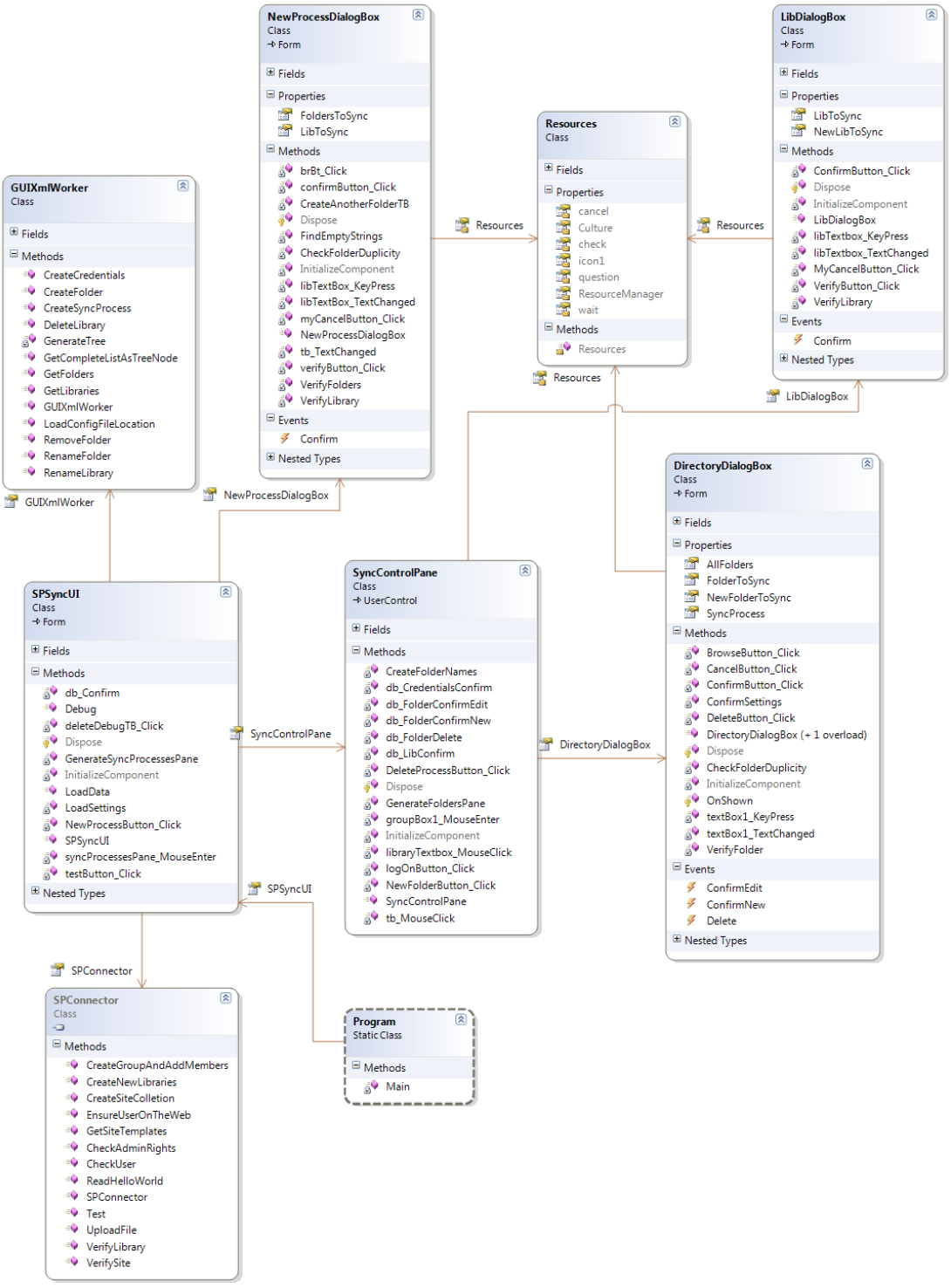


**Figure 19 - SPSyncUI Class Diagram**

### 6.2.3. Project conclusion

The services and application, I have created, synchronize chosen system folders with site libraries, thus work with SharePoint is much more approachable. User doesn't have to spare time going through SharePoint management structures and upload new files or even download them to a

computer for editing and then upload them back. He just opens synchronized folder, edits the file and saves it. My applications will provide all necessary uploading and back up managing services.

## 6.3. BackupViewer Web Part

BackupViewer is a simple Web Part made for managing our backups made by SharePoint File Synchronizer (SPSyncService).

User types actual file site-relative path and the BackupViewer checks, if the site supports backuping, loads settings for the site and finds all backups for the file, if exist. Then shows backup's information like file size, date of last modifying etc.

User can manipulate with backups – download them to his computer, delete them and restore them. When restoring is executed, the backup file and actual file are switched.

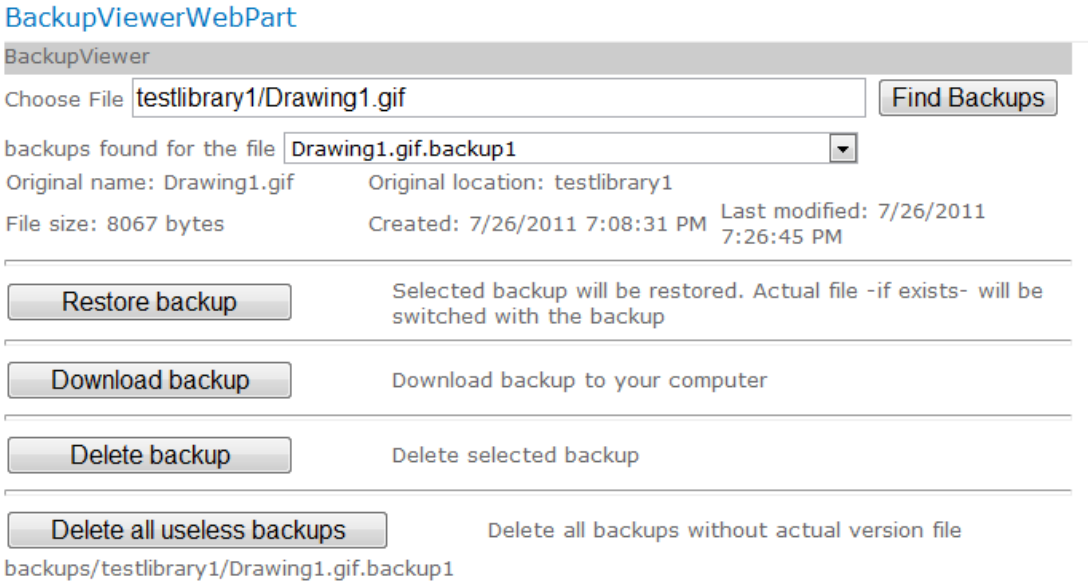


**Figure 20 - BackupViewer Web part**

### 6.3.1. User guide

- Add BackupViewer to your page
- Type site-relative path to the actual file and click on "Find backups" button
- Choose the backup you want to manipulate with.
- Restore/Delete/Download the backup
- Clear backup directory

### 6.3.2. Implementation

When the WebPart is loaded to the page, it checks if backup directory and backup settings file exist on the site. If they don't exist, there is no reason to use BackupViewer. Use SPSyncService application to enable backup support for the site.

When you click on a "Delete all useless backups" button, the BackupViewer goes through whole backup folder and its subfolders and checks whether an actual file for these backups exists. If it does not exist, the backups are probably useless and so they are deleted.

### 6.3.3. Project conclusion

I have developed custom web part using my new knowledge about SharePoint Web Parts and SharePoint API. The Web Part is not just a randomly chosen piece of software, but has a practical use within whole diploma thesis and completes my other projects. You can view and manage backups, which were created by SPSyncService and with this web part backups has practical meaning.

# 7. Conclusion

So this is the end of my diploma thesis. When I started, I didn't know about SharePoint more than a name and few rumors. I knew that SharePoint is widespread in development and business branch and that it had been getting more and more popular, but I didn't know why it had been happening. That changed. I think I can say that I am already familiar with SharePoint – I know what SharePoint really is and why companies deploy SharePoint servers for their employees and projects and I am able to deploy my own SharePoint for myself. I described SharePoint structural basics, I have apprised a reader of the interface and showed how to create a new site collection for your project.

Theory is one thing, but practice counts. I developed 4 programs and web part for 3 projects to prove that I am really able to work with SharePoint and program applications for it and using SharePoint's API. Those applications are not useless programs that would be forgotten and nobody would see them again. They were created to increase a beneficial effect of SharePoint for school projects and all of them altogether form one coherent project.

If you are interested in SharePoint and you want to install your own SharePoint farm, read an appendix of this thesis. There are very useful information for your starts like requirements, licenses and installation procedures.

Thank you for your time.

# 8. Bibliography

[1] Microsoft. Microsoft SharePoint. [Online]. http://technet.microsoft.com/en-us/sharepoint/default

[2] Queensland state archives. (2008, Feb.) Queensland state archives. [Online]. http://www.archives.qld.gov.au/publications/PublicRecordsBriefs/what_is_microsoft_Info_rights_management_200409.pdf

[3] Microsoft. (2010, May) MSDN. [Online]. http://msdn.microsoft.com/en-us/library/ee536171.aspx

[4] Microsoft. (2011, Apr.) MSDN TechNet. [Online]. http://technet.microsoft.com/en-us/library/cc262485.aspx

[5] Thiru Thangarathinam. (2003, Aug.) Developers.com. [Online]. http://www.developer.com/net/asp/article.php/2245511/IIS-and-ASPNET-The-Application-Pool.htm

[6] Microsoft. MSDN. [Online]. http://msdn.microsoft.com/en-us/library/d56de412(v=VS.100).aspx

[7] Microsoft. (2010, May) MSDN. [Online]. http://msdn.microsoft.com/en-us/library/ee554869.aspx

[8] Rico Mariani. (2009, June) MSDN Blogs. [Online]. http://blogs.msdn.com/b/ricom/archive/2009/06/10/visual-studio-why-is-there-no-64-bit-version.aspx

[9] Eric van Glabbeek. (2009, Nov.) sharepoint developer resources. [Online]. http://sharepoint.vanglabbeek.nl/index.php/2009/11/02/sharepoint-foundation-2010-whats-in-a-name/

[10] Microsoft. Microsoft SharePoint. [Online]. http://technet.microsoft.com/cs-cz/sharepoint/ee518670.aspx

[11] Rand Morimoto. (2010, Jan.) Network World. [Online]. http://www.networkworld.com/community/node/56630

[12] Ondřej Pasch, *SharePoint 2010 Praktický průvodce uživatele*.: C Press, 2011.

[13] Microsoft. (2009, Oct.) MSDN Magazine. [Online]. http://msdn.microsoft.com/en-us/magazine/ee309510.aspx

[14] Miloš Sobotka, Jan Lávička Tomáš Kuťěj, *Technologie Microsoft SharePoint 2003 - Implementace, administrace a vývoj*. Brno: CPress, 2006.

[15] Eric White. (2011, 55) MSDN. [Online]. http://blogs.msdn.com/b/ericwhite/archive/2010/05/11/getting-started-building-a-wcf-web-service.aspx

[16] Microsoft. (2010) Compare Microsoft SQL Server Editions. [Online]. http://www.microsoft.com/sqlserver/en/us/product-info/compare.aspx

# 9. Appendix

The appendix contains some interesting information such as licenses, requirements, server preinstall preparations and SharePoint installation that could be useful to everybody who wants to deploy his own SharePoint server.

## 9.1. Licenses and proper version choice

Now when we know the most important technological differences between particular versions, we can decide, which ones we should use for our needs to optimize price and usability. Important is SharePoint usage range whether we have already some server and what configuration it is - namely hardware and software configuration. The cheapest solution couldn't be necessary the most suitable or even legal according to license terms.

First of all we have to decide what kind of server we need to buy. The cheapest option is Web Server. With it we don't pay for number of users using it and thus doesn't matter if 5 or 5000 people access it and we adhere just with number of our server processors. However license terms don't allow us such application. Web server can be used only for front-end web services like a POP3 email service, webs, and web applications and so on. So can be used only for web applications used by internet users and not for in-house purposes (intra-corporate purposes) and that is exactly what we want to do.

With other Windows server versions is important to consult number of CALs (Client Access License) – it is number of users who will utilize the server, get information from it a upload theirs new information or edit existing ones or number of devices, that utilize the server and are utilize by users themselves (even with multiple users).

In the case of older operation system upgrade is necessary to upgrade CALS as well knowing that new licenses have retroactive application but it doesn't work on the other way. For example you can use Windows Server 2008 CALs for Windows Server 2005, but not for Windows Server 2008 R2 and you have to upgrade them to newer version.

SharePoint Foundation 2010 is completely for free and if we choose this possibility and we know it is good enough for our purposes, we don't need to worry with other paid licenses. Otherwise we have to pay for extended SharePoint server 2010.

For proper SharePoint Foundation 2010 function we of course need even database SQL server. For cost reducing we can use Express version, that is free and is fully sufficient for common usage, though there are not advanced administration functions and a database size is reduced to 4GB

(10GB in a case of R2 version – that is good enough for most of small and middle sized companies). And if even this is not enough you still can freely expand your database using BLOB (Binary large object) storage up to total 16 GB.

Problem of SQL Express version is missing Management studio which is not included in installation package, but is very useful and helps you to work with databases, queues, views and so on. But fortunately simplified version of Management studio is legally downloadable for free from Microsoft sites:

| http://www.microsoft.com/downloads/en/details.aspx?FamilyID=08e52ac2-1d62-45f6-9a4a-4b76a8564a2b) |
|---|

If you chose the paid SQL Server version, you have to count with a need to buy CAL licenses for each user both for SharePoint server itself (and you can't shirk from it), and for every user who is going to use it.

All these payments are single-use and it is not need to pay regularly.

## 9.2. Requirements

Hardware and software requirements: [4]

There are several installation scenarios, with specific requirements from simple fore-end installation (all-in-one-machine) for small companies, up to large server farms for big corporations.

*Server requirements (web servers, application servers, all-in-one servers):*

64bit processor architecture, 4GB RAM for development and 8GB RAM for deployment on one machine or several in farm and 80GB space on disk.

*Database requirements:*

64bit processor architecture with at least 4 cores, 8GB RAM minimum and 80GB hard drive space.

*Software requirements:*

SQL Server 2005 with SP3 and cumulative update 3 or
SQL Server 2008 with SP1 and cumulative update 2 or 5 or later (3 a 4 are strictly not recommended) or SQL Server 2008 R2

It always best to use newest R2 version, if there are no other technical limits.

*Client requirements:*

Browser (Internet Explorer 9 or newest FireFox with some limitations)

## 9.3. Server pre-install preparations

Setting up Active Directory Domain Services (ADDS):

- For installation is necessary to configure DNS server to locate our farm in server domain
- Then we have to create new forest in DNS for our domain or add to existing one
- We have to create new root domain FQDN (fully qualified domain name) for our new forest. I have used FQDN: server.testing.com for my needs
- Next we have to set up a functionality range of our new forest and that mean for all domains in it. We can choose one of several versions. It is because of compatibility needs in some cases with older systems:
    - Windows 2000 with full ADDS support for Windows 2000
    - Windows Server 2003 with the same functionality like the older one with few more features like Linked-value replication, which improves the replication of changes to group memberships, More efficient generation of complex replication topologies by the KCC and Forest trust, which allows organizations to easily share internal resources across multiple forests.
    - Windows Server 2008 forest version does not bring any new functionality, but allow as automatic communication on Windows Server 2008 domain level.
    - Windows Server 2008 R2 forest version adds a Bin functionality (if enabled) so any deleted object can be restored again. Plus adds automatic communication on Windows Server 2008 R2 domain level.

It is all we need. DNS Server is set up. Now we can create a domain for our machine and add it to the new forest.

Of course we need application and IIS server roles enabled on the server. We can do it manually or let it make automatic SharePoint installation prerequisites installation.

## 9.4. SharePoint installation

SharePoint installation is very simple. Installation program can check, download and set all prerequisites up if needed.

During installation configuration is possible to choose, if we want to have whole server with databases on one machine, which is cheaper option and suitable mostly for individuals or smaller or middle size companies or we want separated application and data parts of server thus the loading is spread over more machines (front-end and back-end option when front-end accessible application SharePoint server is on one or several connected machine and database SQL server on other one). That comes useful when we expecting larger amount of access to a server from much larger amount of users. Then is necessary to join SharePoint to already existing web farm or create a new one.

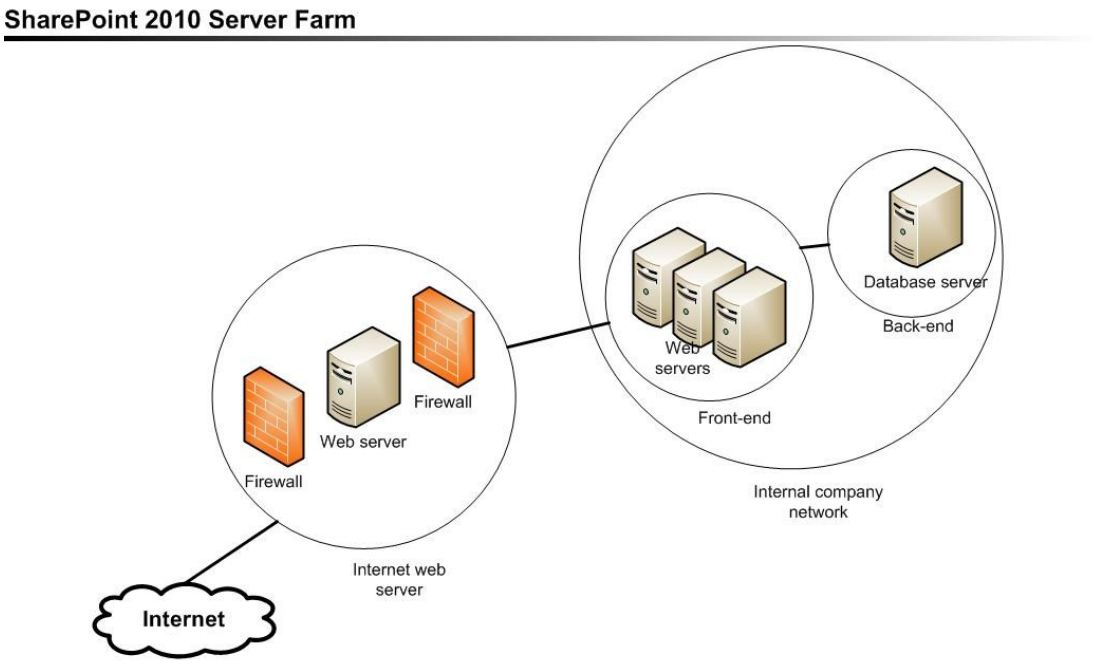


**Figure 21 - SharePoint 2010 Server farm diagram**

SharePoint needs connection to database server and express version of SQL server is good enough for functioning. In the case that SQL service doesn't work, is possible that it is due to more tight access rights compared to desktop version (for example Windows 7). You can fix the problem with changing of privileges or changing a logon account the service is using – From Network Service (on default) to Local system. And that is including SQL Server Agent, which the SQL Service depends on. At the end you have to set a password to server farm up.

Than you have to create a Central Administration Web with SharePoint Products Configuration Wizard.

You have to set up SQL Server name and create a new database with a name SharePoint_Config (on default) and login to it, user in complete format with including domain in a form DOMAIN\User_name and create a password to secure SharePoint product farm, which will be

demanded when configuration is going to be changed or login like a SharePoint Farm Administrator. The last thing is to configure basic SharePoint Central Administration site settings – communication port and authentication protocol (NTLM or Kerberos).